

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

Contribution à la conception d'un système informatique d'aide à l'enseignement de la physiologie cardiaque : élaboration d'un modèle qualitatif du cœur gauche selon les principes de la programmation logique

Henrioul, Anne; Warnier, Dominique

*Award date:*  
1986

*Awarding institution:*  
Université de Namur

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX - NAMUR

INSTITUT D'INFORMATIQUE

CONTRIBUTION A LA CONCEPTION  
D'UN SYSTEME INFORMATIQUE D'AIDE A  
L'ENSEIGNEMENT DE LA PHYSIOLOGIE  
CARDIAQUE :

ELABORATION D'UN MODELE QUALITATIF  
DU COEUR GAUCHE SELON LES PRINCIPES  
DE LA PROGRAMMATION LOGIQUE

ANNE HENRIOUL  
DOMINIQUE WARNIER

ANNEE ACADEMIQUE 1985-1986

MEMOIRE PRESENTE EN VUE DE  
L'OBTENTION DU GRADE DE LICENCIE  
ET MAITRE EN INFORMATIQUE

- "bon" de passer au qualitatif! /  $\exists$  l'ys approche quantitative
- génération automatique des règles qualitatives

le quantitatif élimine autrement de l'information que le qualitatif.  
 $\Rightarrow$  il n'y en a pas une "juste" ?!

grand  $\div$  ce dont on parle  
 mais aussi  $\div$  au moment auquel on en parle

+ naturel de représenter une f<sup>on</sup> numérique en lisp qu'en Prolog  
 logique

conclusion de  
 comparaison  
 Prolog - lisp.  
 a effr cadit  
 parle s/ une  
 in présentation

lien qualitatif + graphique  
 modèle linéaire? (aux Capacités verbales)

Nous tenons tout d'abord à exprimer notre reconnaissance à Monsieur J. BARRETO qui a proposé ce mémoire et qui nous a permis, par ses conseils et critiques, de mener ce travail à son terme.

Nous remercions également Madame NOIRHOMME-FRAITURE pour l'aide qu'elle nous a prodiguée tout au long de ce travail.

Nous exprimons notre gratitude à Monsieur LEFEVRE, assistant à l'UCL en Woluwe, qui nous a enseigné les principes de la physiologie cardiaque.

Enfin, nous adressons nos remerciements à toutes les personnes qui, de près ou de loin, ont participé à la réalisation de ce mémoire.



# TABLES DES MATIERES

CHAPITRE 1	INTRODUCTION	1
CHAPITRE 2	DETERMINATIONS DES BESOINS DE L'UTILISATEUR ET DES OBJECTIFS A ATTEINDRE	7
2.1	INTRODUCTION	7
2.2	LES BESOINS	8
2.3	LES OBJECTIFS	8
CHAPITRE 3	NOTIONS DE PHYSIOLOGIE CARDIO-VASCULAIRE	10
3.1	ORGANISATION GENERALE DU SYSTEME CARDIO-VASCULAIRE	10
3.2	LE COEUR	10
3.2.1	PRESENTATION GENERALE	10
3.2.2	DESCRIPTION DES COMPOSANTS DU COEUR GAUCHE ET DE L'AORTE	10
3.3	LE CYCLE CARDIAQUE	13
3.3.1	LE PHENOMENE DE LA CONTRACTION	13
3.3.2	LES QUATRE PHASES DU CYCLE CARDIAQUE	14
3.4	FACTEURS DE PERFORMANCE DU COEUR	17
3.4.1	FACTEURS DETERMINANT LE REMPLISSAGE DU VENTRICULE	20
3.4.2	FACTEURS DETERMINANT LA PERFORMANCE DU VENTRICULE	20
3.5	PATHOLOGIES CARACTERISTIQUES	21
3.5.1	ARTERIOSCLEROSE	21
3.5.2	STENOSE ET INSUFFISANCE AORTIQUE	22
3.5.3	STENOSE ET INSUFFISANCE MITRALE	23



CHAPITRE 4	MODELES VENTRICULAIRES	24
4. 1	INTRODUCTION	24
4. 2	MODELE HYDRAULIQUE	25
4. 3	MODELE ELECTRIQUE	28
4. 3. 1	PRESENTATION DU MODELE	28
4. 3. 2	EQUATIONS TIREES DU MODELE	30
CHAPITRE 5	ACQUISITION DES CONNAISSANCES	33
5. 1	INTRODUCTION	33
5. 2	VARIABLES ET PARAMETRES DU SYSTEME	34
5. 2. 1	LES VARIABLES	34
5. 2. 2	LES PARAMETRES	35
5. 3	REGLES DE RAISONNEMENT DETERMINANT L'ETAT DU SYSTEME	35
5. 3. 1	CALCUL DES PRESSIONS	35
5. 3. 2	CALCUL DES VOLUMES	36
5. 3. 3	DETERMINATION DE LA POSITION DES VALVES	36
5. 3. 4	CALCUL DES DEBITS	36
5. 3. 5	CALCUL DE LA VARIATION DES VOLUMES	38
5. 4	DESCRIPTION GENERALE DES PHASES DU CYCLE CARDIAQUE VIA LES VARIABLES DU SYSTEME	39
5. 4. 1	PHASE DE REPOS	39
5. 4. 2	PHASE DE CONTRACTION ISOVOLUMETRIQUE	39
5. 4. 3	PHASE DE CONTRACTION EJECTION	40
5. 4. 4	PHASE DE RELAXATION	40
CHAPITRE 6	RAISONNEMENT QUALITATIF	42
6. 1	INTRODUCTION	42
6. 2	TRAVAUX EXISTANT DANS LE DOMAINE DU RAISONNEMENT QUALITATIF	44

6.3	CALCUL DES VALEURS DES VARIABLES A L'INSTANT COURANT	47
6.3.1	CHOIX DES VALEURS QUALITATIVES	47
6.3.2	OPERATIONS QUALITATIVES	54
6.4	CALCUL DES VALEURS DES VARIABLES A L'INSTANT SUIVANT	60
6.4.1	PRINCIPES	60
6.4.2	CHOIX DES ETAPES ET LONGUEURS DES INTERVALLES	62
6.4.3	PROPAGATION DES VARIABLES D'ETAT	63
CHAPITRE 7	IMPLEMENTATION MICRO-PROLOG	69
7.1	INTRODUCTION	69
7.2	PRESENTATION DES PRINCIPALES CARACTERISTIQUES DU PROGRAMME MICRO-PROLOG	69
7.2.1	PRINCIPE GENERAL D'UNE EXECUTION	69
7.2.2	CONTENU DE LA BASE DE CONNAISSANCES	73
7.2.3	MODIFICATION DE LA BASE DE CONNAISSANCES	78
7.2.4	LES ARTIFICES DE PROGRAMMATION NECESSAIRES A L'OBTENTION DE REPONSES CORRECTES DE LA PART DU SYSTEME	80
7.2.5	PROBLEMES INHERANTS A LA PROGRAMMATION LOGIQUE	84
7.3	CRITIQUE DE LA PROGRAMMATION LOGIQUE ET DE MICRO-PROLOG	85
7.3.1	CRITIQUE DE LA PROGRAMMATION LOGIQUE	86
7.3.2	CRITIQUE DU LOGICIEL MICRO-PROLOG	87
CHAPITRE 8	COMPARAISON PROLOG-LISP	89
8.1	POURQUOI UNE ETUDE COMPARATIVE ENTRE PROLOG ET LISP	89
8.2	PRINCIPES DE BASE DE PROLOG ET DE LISP	90



8.3 DIVERGENCES OBSERVEES ENTRE LISP ET PROLOG DANS LE CADRE DE NOTRE SYSTEME	90
8.3.1 PROCEDURES PROLOG ET FONCTIONS LISP	91
8.3.2 BACKTRACKING PROLOG	92
8.3.3 OUTPUTS	93
8.3.4 ARGUMENTS EN ENTREE ET EN SORTIE	94
8.3.5 FORMALISATION DES CONNAISSANCES	95
8.3.6 LISIBILITE	95
8.3.7 TABLES ET CALCULS QUALITATIFS	96
8.3.8 MISE A JOUR DE LA BASE DE CONNAISSANCES	98
8.3.9 ORDONNANCEMENT DES CLAUSES	98
8.4 CONCLUSION	99
CHAPITRE 9 CONCLUSION	100
BIBLIOGRAPHIE	105
ANNEXE 1 TABLES DES OPERATIONS QUALITATIVES	
ANNEXE 2 EXEMPLE D'EXECUTION DES PROGRAMMES PROLOG (INITIALISATION ET PROGRAMME PRINCIPAL)	
ANNEXE 3 PROGRAMMES PROLOG	
ANNEXE 4 PROGRAMME LISP	



## CHAPITRE 1

### INTRODUCTION

Dans le but de mieux faire comprendre les principaux phénomènes de la physiologie cardiaque aux étudiants en médecine de Louvain en Woluwe, une présentation de modèles cardio-vasculaires a été intégrée dans leur programme de cours. Ces modèles permettent une meilleure visualisation des mécanismes fondamentaux du fonctionnement du coeur. Les résultats d'expérimentations sur ces modèles sont, jusqu'à présent, justifiés et commentés par un assistant.

Dans le cadre de cette étude de modèles cardio-vasculaires, il est apparu intéressant d'utiliser l'ordinateur comme un outil d'aide à l'enseignement. La tâche assignée au système informatique à développer à long terme serait assez complète. D'une part, ce système simulerait le fonctionnement du coeur sur plusieurs cycles cardiaques; d'autre part, il serait capable de justifier à l'utilisateur, en l'occurrence l'étudiant, tout résultat observable de la simulation.

Le développement d'un projet d'une telle envergure n'est cependant pas réalisable en quelques mois. Il est donc nécessaire de définir un objectif plus modeste qu'il sera possible d'atteindre dans de plus brefs délais. Il est apparu plus raisonnable, bien que toujours aussi intéressant, de concevoir un système qui simulerait et expliquerait uniquement le comportement du coeur gauche (oreillette gauche, ventricule gauche, aorte). La restriction que nous opérons à ce niveau porte principalement sur la dimension de l'objet du réel à modéliser. La nature même des problèmes à résoudre afin d'obtenir un système d'aide à l'enseignement satisfaisant n'en est pas pour autant modifiée.

Il est à présent nécessaire, de caractériser un peu plus le système attendu. Celui-ci peut, en effet, présenter quelques différences par rapport aux programmes d'E.A.O. développés jusqu'ici. Il est bien entendu que le système doit simuler le fonctionnement du coeur gauche et interpréter les résultats



fournis par la simulation. Cependant, n'oublions pas que ce système s'adresse à de futurs médecins qui, grâce à lui, doivent approfondir et concrétiser leurs connaissances en physiologie cardiaque. Il semble alors primordial de développer un outil qui propose aux étudiants une simulation du comportement du coeur gauche qui s'inscrive effectivement dans leur mode habituel de raisonnement.

D'une manière générale, lorsqu'un homme doit prendre des décisions dans des environnements complexes ou incertains, le raisonnement qui le guide est approximatif [1]. Souvent un tel raisonnement est plutôt de nature qualitative que quantitative et sort du domaine d'applicabilité d'une logique classique rigoureuse. En particulier en ce qui concerne le fonctionnement du coeur gauche, nous pouvons remarquer que pour caractériser l'état ou l'évolution des composants cardiaques il est courant d'utiliser des grandeurs qualitatives. Les physiologistes parleront notamment d'une augmentation rapide et importante de la pression ventriculaire gauche suite à une contraction; du faible débit sanguin vers le réseau périphérique (le corps) lorsque la pression aortique est moyenne, et ceci sans introduire de grandeurs quantitatives dans leur raisonnement.

Afin d'obtenir un système qui simule le coeur gauche et qui justifie adéquatement la démarche suivie pour obtenir les résultats qualitatifs qu'il délivre, il semble nécessaire de résoudre deux problèmes. Tout d'abord, il s'agit de développer un modèle qualitatif du fonctionnement du coeur gauche. La logique des prédicats du premier ordre transposée sous la forme de clauses de Horn constitue l'outil de base de la modélisation. Ensuite, un mécanisme explicatif doit être intégré au modèle afin que le système puisse justifier les résultats de la simulation.

L'intérêt de la réalisation de l'outil d'aide à l'enseignement que nous proposons peut se situer à plusieurs niveaux.

Tout d'abord, ce système offrira à chaque étudiant la possibilité d'une approche plus personnelle de son étude. Il pourra savoir s'il a correctement compris une leçon en se référant aux explications fournies par le programme et il pourra revoir, s'il le souhaite, certains points moins bien assimilés. Toujours dans le cadre de l'enseignement, le système pourra éventuellement aider le professeur à mieux structurer son cours et à le rendre plus complet.



Du point de vue conceptuel, l'intérêt du travail se porte essentiellement sur l'approche et le traitement du raisonnement qualitatif que toute personne est amenée à faire lorsqu'elle étudie des systèmes de manière intuitive.

Enfin, en utilisant un outil de programmation logique pour formaliser notre problème nous pouvons aborder de plus près les avantages et les inconvénients de ce type d'implémentation.

Avant de s'attacher effectivement à notre problème particulier, nous présentons brièvement quelques travaux apparentés aux domaines de l'enseignement assisté par ordinateur et du raisonnement qualitatif sur des systèmes.

L'introduction progressive de l'informatique dans le domaine de l'éducation fut favorisée par l'apparition sur le marché de micro-ordinateurs performants et de prix accessibles.

Certains auteurs qui se sont plus particulièrement intéressés aux apports de l'ordinateur dans l'enseignement supérieur considèrent qu'il existe cinq façons distinctes de l'utiliser dans un environnement académique [2] [3].

Premièrement, les ordinateurs peuvent simplement constituer un objet d'étude dans le cadre de l'enseignement et de la recherche en informatique.

Deuxièmement, ils peuvent être utiles pour l'administration courante et la gestion de l'établissement scolaire.

Troisièmement, ils peuvent intervenir dans des opérations de traitement de texte, d'analyse de document, et dans la gestion de bases de données.

Quatrièmement, ils peuvent être utilisés comme outil d'aide à l'enseignement de diverses matières.

Dans ce dernier cas, il est possible d'envisager quelques utilisations académiques futures de l'ordinateur. Celui-ci pourrait jouer le rôle d'un simulateur de phénomènes physiques ou sociaux, trop complexes à comprendre à partir de formules mathématiques. En effet, il arrive souvent que les étudiants connaissent les formulations mathématiques qui modélisent les phénomènes à étudier. Cependant, beaucoup d'élèves affirment qu'il est très difficile de passer d'une formulation mathématique abstraite à



une compréhension intuitive réelle des phénomènes. L'ordinateur pourrait en quelque sorte aider les étudiants à concrétiser leurs connaissances.

Il pourrait encore être envisagé comme un instrument de laboratoire. Le principal avantage à dégager de cette approche est qu'elle permet l'élimination des tâches ennuyeuses et inutiles qui sont toujours associées aux exercices de laboratoire. L'étudiant parvient alors à mieux se concentrer sur les phénomènes physiques importants à étudier.

Enfin, l'ordinateur pourrait participer au travail de l'enseignant. Actuellement, les recherches menées en Intelligence Artificielle orientent ce type d'applications vers des expériences plus intéressantes et encourageantes pour les étudiants. De nouveaux systèmes appelés "Expert Tutor" permettent aux étudiants d'abandonner une attitude passive pour devenir le premier agent du dialogue avec la machine. Dans de tels systèmes, c'est avant tout l'étudiant qui tente de résoudre les problèmes et l'ordinateur peut le conseiller sur demande.

En ce qui concerne l'apprentissage à l'U.C.L. des mécanismes du fonctionnement du coeur, un micro-ordinateur est utilisé comme outil pédagogique. Il permet aux étudiants d'observer des graphiques caractérisant l'activité cardiaque commentée par un assistant. Les graphiques sont élaborés sur base des solutions des systèmes d'équations qui définissent des modèles cardiovasculaires. Ceux-ci furent développés dès 1981 par une équipe du laboratoire de physiologie cardio-vasculaire de l'U.C.L.. Cette équipe travaillait, dans le cadre d'un projet F.D.S. (Fond de recherches scientifiques), au développement d'un software de simulation qui serait utilisé dans le domaine de l'E.A.O. [4] [5] [6] [7] [8] [9]. Par la suite, il s'est avéré intéressant de concevoir un système plus complet qui proposerait une simulation qualitative du fonctionnement du coeur. Il devrait également justifier et commenter tout résultat de la simulation en des termes significatifs aux étudiants en médecine.

Notre système constitue en quelque sorte un élément de la chaîne de conception de cet outil puisqu'il propose une simulation qualitative du fonctionnement du coeur gauche. Cependant, actuellement, il n'offre pas la possibilité d'expliquer la démarche suivie pour obtenir les résultats de la simulation. Ce mécanisme explicatif devra être réalisé par la suite.



Afin de réaliser un modèle qualitatif du fonctionnement du coeur gauche sur plusieurs cycles cardiaques, nous nous sommes intéressées aux diverses recherches effectuées dans le domaine du raisonnement qualitatif. Rappelons à ce propos, qu'un raisonnement de type qualitatif s'appuie généralement sur des domaines d'informations incertaines. C'est pourquoi nous considérons également des méthodes de traitement du raisonnement sur des données imprécises.

L'idée maîtresse qui paraît dans la plupart des méthodes dont nous avons pris connaissance est de proposer des techniques de représentation de systèmes qui permettent d'acquérir une meilleure compréhension de phénomènes du monde réel. Nous avons pu remarquer que ces méthodes se différencient quelque peu en fonction du type de phénomènes à étudier. En particulier, s'il s'agit de raisonner sur des systèmes physiques, certains chercheurs proposent de développer une physique qualitative qui serait mieux adaptée qu'une physique standard trop mathématique [10] [11] [12]. Par contre, s'il s'agit de raisonner sur des systèmes plus généraux, d'autres techniques peuvent être utilisées [13]. Citons par exemple, une des approches les plus répandues dans ce domaine qui se base sur des inférences de certitude parallèles. Lorsqu'une inférence est incertaine, elle peut être augmentée par une autre dont le but est de calculer la certitude de la première. Des renseignements complémentaires sur les méthodes que nous avons abordées sont proposés dans le chapitre 6 de ce document. Toutefois nous pouvons dès à présent indiquer que nous n'avons pas opté pour l'une ou l'autre de ces techniques, en particulier dans la mesure où elles ne nous semblaient pas appropriées au type de traitement que nous souhaitions réaliser.

Comme nous l'avons déjà mentionné, notre approche qualitative se limite à l'étude du coeur gauche. Nous proposons un modèle qualitatif qui inclut en partie le raisonnement que tient un enseignant lorsqu'il expose à des étudiants le comportement du coeur gauche. Cependant le modèle établi est incomplet dans la mesure où il n'envisage aucune possibilité de diagnostics. Notre système offre, toutefois, la possibilité d'observer en détail le fonctionnement du coeur gauche sur plusieurs battements cardiaques. Et les valeurs des variables d'état du système tels que, par exemple, la pression dans le ventricule gauche ou le volume de sang dans l'aorte, sont exprimées dans le domaine qualitatif significatif à l'étudiant. Quant à l'interprétation ou la justification de ces valeurs de variables, notre système est limité puisque nous n'avons pas implémenté le mécanisme explicatif du



raisonnement.

L'implémentation de notre système est réalisée en Micro-Prolog malgré les imperfections qu'il présente [15]. Nous signalons à ce propos que l'un des intérêts de ce travail porte sur la démarche méthodologique à adopter pour traiter un problème des domaines de l'E.A.O. et du raisonnement qualitatif. Nous n'avons nullement l'intention de rechercher LE formalisme qui permettrait une implémentation optimale du système. D'autre part, nous désirions aborder la programmation logique et à l'époque où nous commençons ce travail Micro-Prolog était le seul outil dont nous disposions.

Pour terminer voici en quelques mots la démarche que nous avons suivie pour élaborer le modèle qualitatif du fonctionnement du coeur gauche. Elle peut être décomposée en différentes étapes qui feront chacune l'objet d'un chapitre dans la suite de ce document. Après avoir pris connaissance des besoins du département de physiologie cardiaque de Woluwé nous devions avant toute autre chose acquérir des notions physiologiques. Après cette étape, différents modèles du coeur nous étaient présentés. Nous disposions alors de suffisamment d'informations pour élaborer par déduction, des règles générales de raisonnement. Nous avons réfléchi également à la façon d'intégrer à ces règles générales les dimensions qualitative et temporelle nécessaires à la modélisation d'une activité cardiaque sur plusieurs cycles. Nous avons alors formalisé les connaissances et les règles de raisonnement du domaine étudié à l'aide de Micro-Prolog bien que, comme nous l'avons déjà fait remarquer, ce langage n'est sans doute pas le plus adéquat. En tenant compte de ce fait, nous avons reformalisé le problème à l'aide de IQ-LISP, un outil mis à notre disposition beaucoup plus tard. Nous avons enfin tenté une comparaison sommaire de ces deux langages principalement du point de vue de la programmation.



## CHAPITRE 2

### DETERMINATION DES BESOINS ET DES OBJECTIFS A ATTEINDRE

#### 2.1 INTRODUCTION

---

En complément d'un cours de cardiologie donné à l'Université Catholique de Louvain en Woluwe, des travaux pratiques sont proposés aux étudiants. L'une de ces séances d'exercices présente via un micro-ordinateur, des modèles cardiovasculaires. Ceux-ci permettent aux étudiants de mieux percevoir et de bien assimiler les mécanismes fondamentaux qui régissent une activité cardiaque complète. Les résultats de diverses expérimentations sur ces modèles sont repris dans des graphiques et sont commentés par un assistant.

Dans le but d'améliorer la qualité de ces travaux pratiques, il est envisagé de modifier l'utilisation de l'ordinateur faite jusqu'ici. D'une manière générale, il est question de concevoir un système informatique qui soit capable de simuler le fonctionnement du coeur en activité et de commenter les résultats de cette simulation. Cependant, il est souhaité que le système propose une simulation et des justifications du comportement du coeur qui soient significatives aux étudiants. En d'autres mots, il s'agit que le système donne l'impression d'exposer et d'expliquer l'activité cardiaque tout comme le fait actuellement l'assistant.

Notre travail vient, en quelque sorte, s'intégrer dans le processus d'élaboration de ce système et constitue une première étape de la démarche de conception. Nous envisageons, pour ce faire, l'étude du comportement du coeur gauche isolé. Nous précisons ci-dessous les différents besoins à satisfaire ainsi que les objectifs à atteindre pour réaliser un système qui corresponde aux mieux aux besoins retenus et à l'activité du coeur gauche isolé.



## 2.2 LES BESOINS

---

Le principal souhait de l'assistant qui encadre actuellement les travaux pratiques dont nous avons dit quelques mots précédemment, est d'avoir à sa disposition un système d'E.A.O. qui permette aux étudiants en médecine de connaître le comportement du coeur gauche sur plusieurs cycles cardiaques. Cet apprentissage doit être facilité par une présentation plus intuitive, plus qualitative de l'activité du coeur isolé. L'étudiant disposera également de la possibilité de s'attacher à l'étude d'un coeur sain ou d'un coeur malade. Dans tous les cas, il pourra interroger le système pour connaître l'état ou les caractéristiques du coeur isolé à l'instant courant d'un cycle ou après évolution dans le temps de son comportement.

## 2.3 LES OBJECTIFS

---

L'objectif fondamental à atteindre dans ce travail est de concevoir un système paramétrable qui simule qualitativement le fonctionnement du coeur gauche sur plusieurs cycles cardiaques. L'adjectif paramétrable signifie que les conditions initiales d'une simulation qui définissent un coeur sain peuvent être modifiées pour étudier certaines pathologies. D'autre part, le terme qualitativement signifie que tout résultat observable de la simulation n'est pas représenté par une valeur numérique mais qualitative. C'est pourquoi, il sera courant d'observer une "grande" pression dans l'aorte ou un "petit" volume de sang dans le ventricule gauche.

Nous procéderons en trois étapes pour mettre au point ce système. Tout d'abord nous développerons un petit sous-système qui permette de représenter le coeur gauche à un instant donné du cycle cardiaque. L'état du coeur à cet instant sera déterminé par les valeurs qualitatives et le sens de variation des caractéristiques de ses composants. Par la suite, le sous-système sera complété de manière à représenter l'évolution dans le temps du comportement du coeur gauche. Il s'agira essentiellement de faire progresser les valeurs des principales caractéristiques des composants du coeur gauche. Enfin, nous envisagerons la possibilité de modifier les conditions initiales de la simulation afin de

permettre l'étude de certaines affections du coeur sur son comportement.



## CHAPITRE 3

### NOTIONS DE PHYSIOLOGIE CARDIO-VASCULAIRE

#### 3.1. ORGANISATION GENERALE DU SYSTEME CARDIO-VASCULAIRE

---

Le sang circule unidirectionnellement dans tout le corps. A la sortie des poumons, le sang oxygéné conduit par les veines pulmonaires rentre successivement dans l'oreillette gauche et le ventricule gauche pour y être éjecté dans l'artère aortique. Il irrigue alors le corps par un échange d'oxygène ( $O_2$ ) qu'il possède en suffisance contre de l'acide carbonique ( $CO_2$ ) contenu dans les fibres de chaque organe. Le sang riche en  $CO_2$  est collecté par les veines qui le transportent dans l'oreillette et le ventricule droit d'où il sera chassé dans l'artère pulmonaire menant aux poumons. Il peut ainsi se réoxygéner par échange de  $CO_2$  et oxygène et le cycle recommence. Cette boucle circulatoire est représentée à la figure 3.1. [16] [17] [18]

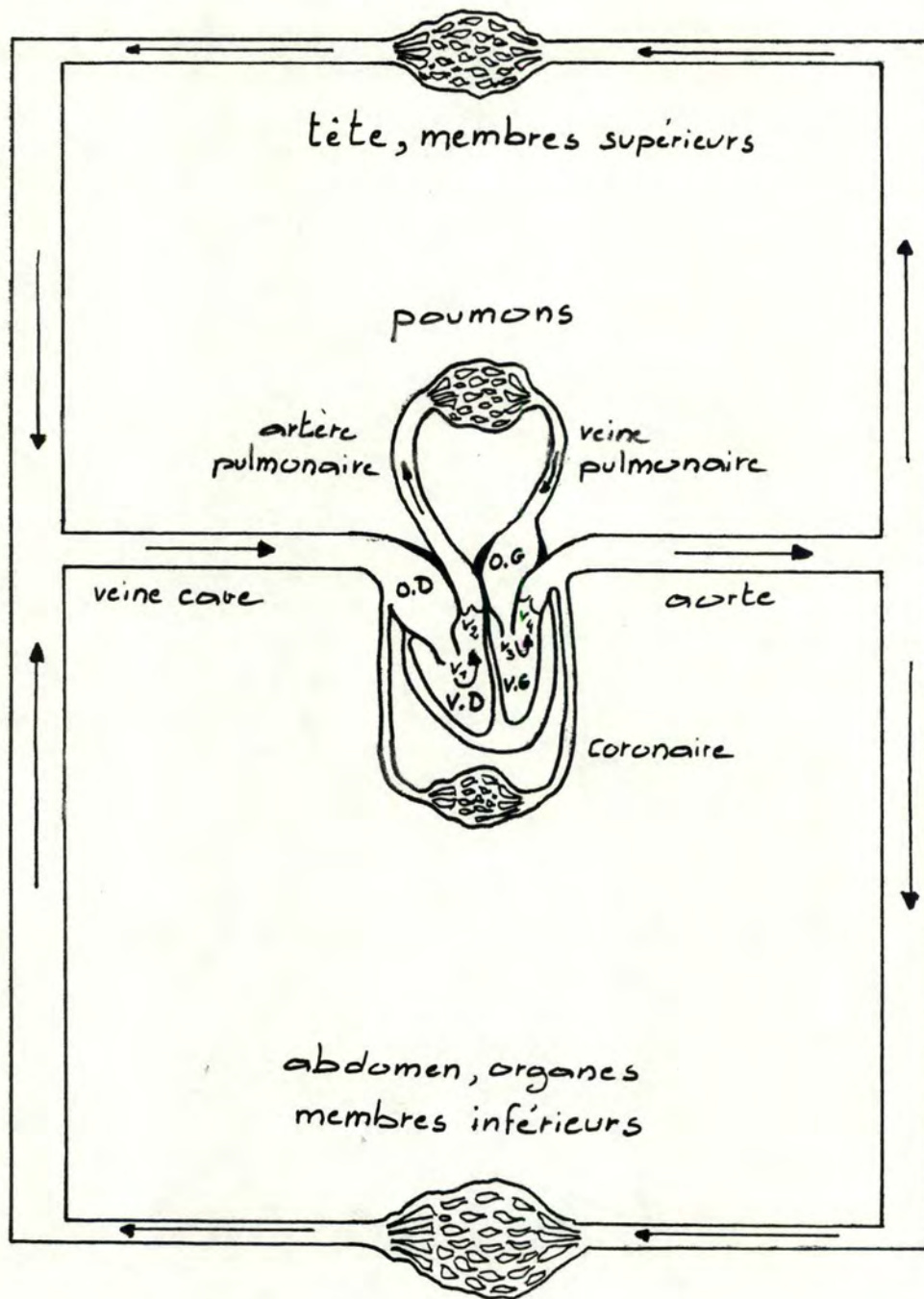
Notre étude se portera uniquement sur le fonctionnement de la partie gauche de la boucle circulatoire, c'est à dire sur l'oreillette gauche, le ventricule gauche et l'aorte.

#### 3.2. LE COEUR

---

##### 3.2.1. PRESENTATION GENERALE

En physiologie cardiaque, il est courant de parler de deux coeurs, un droit et un gauche, assimilables chacun à une pompe aspirante par son oreillette, foulante par son ventricule.



O.D : oreillette droite  
 V.D : ventricule droit  
 O.G : oreillette gauche  
 V.G : ventricule gauche

$v_1$  : valve tricuspide  
 $v_2$  : valve pulmonaire  
 $v_3$  : valve mitrale  
 $v_4$  : valve aortique

Figure 3.1. : Boucle Circulatoire



L'oreillette gauche se vide dans le ventricule gauche par un orifice contenant une valve appelée valve mitrale. L'écoulement du sang rendu possible par la différence de pression régnant dans les deux chambres, est accéléré par une contraction du myocarde auriculaire. Lorsque l'oreillette est vide, la valve mitrale se ferme pour éviter le reflux et le muscle auriculaire se relâche. Le ventricule se contracte à son tour et chasse le sang dans l'artère aortique via une seconde valve (valve aortique). Ce phénomène est appelé systole ventriculaire. Une fois le ventricule débarrassé de son contenu, il se relâche et la valve aortique se ferme (diastole ventriculaire).

### 3.2.2. DESCRIPTION DES COMPOSANTS DU COEUR GAUCHE ET DE L'AORTE

Le coeur gauche comprend 4 éléments (oreillette gauche, ventricule gauche, valve aortique et valve mitrale) auxquels nous ajouterons l'aorte pour les besoins de notre étude.

#### 1. Oreillette gauche :

Chambre musculaire d'un volume réduit par rapport au ventricule. Son rôle consiste à collecter du sang pour assurer un débit régulier vers le ventricule. Sa contraction, bien que n'engendrant que de faibles pressions, assure un surplus de remplissage et permet au ventricule d'atteindre son volume maximal.

#### 2. Ventricule gauche :

Chambre cardiaque communiquant avec l'oreillette par la valve mitrale et avec l'aorte par la valve aortique. Il possède deux caractéristiques qui lui valent d'être l'élément le plus important du coeur : une élasticité et une contractilité importante. Le ventricule est un sac souple pouvant contenir un volume assez important de sang. Son élasticité au repos lui permet de se remplir beaucoup sans augmenter fortement sa pression. Il est capable également de se contracter avec force et atteindre des pressions assez importantes indispensables pour l'éjection de son contenu dans l'aorte.



### 3. Valves (mitrale et aortique)

Membranes passives et très mobiles permettant un passage unidirectionnel du liquide sanguin. La valve s'ouvre quand la pression en amont est plus élevée que la pression en aval et se ferme dans le cas contraire. Une fois ouverte, elle n'offre, dans un système physiologiquement normal, qu'une résistance négligeable au passage du flux, permettant ainsi une éjection ou un remplissage performants. Inversement, la résistance d'une valve peut être considérée comme infinie lorsqu'elle est fermée. Elle évite ainsi tout reflux possible.

### 4. Aorte :

Artère débouchant du ventricule gauche. Sa fonction est de diriger le sang vers les petits capillaires du réseau périphérique (le corps).

Les parois de l'artère sont plus souples que celles du ventricule au moment de sa contraction mais elles sont suffisamment raides pour atteindre des pressions très élevées au moment de l'éjection du fluide sanguin dans l'aorte.

## 3.3 LE CYCLE CARDIAQUE

---

### 3.3.1. LE PHENOMENE DE LA CONTRACTION

#### DESCRIPTION DES FIBRES DU COEUR

Les parois du coeur sont composées de faisceaux de fibres myocardiques, du nom de myofibrilles, entremêlées de façon très complexe. Les micrographies électroniques révèlent que ces myofibrilles sont constituées de filaments encore plus petits dont les uns sont plus épais que les autres (filaments fins d'actine et filaments épais de myosine). Ces deux lots de filaments sont reliés par une succession de petits "ponts d'union" qui jouent un grand rôle dans la contraction musculaire (figure 3.2). La contraction musculaire est considérée comme un processus au cours duquel ces ponts se fixent en des endroits particuliers des molécules d'actine et pivotent jusqu'à une nouvelle position, se libèrent et se fixent en un nouveau point. Le processus peut être comparé aux mouvements des rames se trouvant



de chaque côté d'un canot; les rames plongent dans l'eau, dépen-  
sent une certaine énergie en avançant et se préparent ensuite  
pour un nouveau coup. Les forces de propulsion qui s'exercent  
aux extrémités des filaments aboutissent à des déplacements qui  
se font en un sens opposé si bien que le chevauchement des deux  
types de filaments augmente au cours de la contraction (figure  
3.3). L'énergie nécessaire au fonctionnement de ce mécanisme est  
fournie par une substance chimique particulière appelée l'adéno-  
sine triphosphate (ATP).[16]

#### INITIALISATION DE LA CONTRACTION: COUPLAGE EXCITATION-CONTRACTION

Le glissement entre filaments d'actine et myosine est  
causé par des forces attractives développées entre eux-ci.  
Quand une différence de potentiel se crée entre l'extérieur et  
l'intérieur de la fibre musculaire, il s'effectue un passage  
forcé d'un grand nombre d'ions de Calcium vers l'intérieur de la  
fibre. Ces ions de Calcium activent les forces attractives entre  
filaments et la contraction commence.

L'apparition d'une différence de potentiel est due à  
une stimulation électrique provoquée par l'entremise de fibres  
nerveuses attachées aux fibres musculaires. Le point de liaison  
se situe au milieu de la fibre musculaire. Ainsi, le potentiel  
d'action se répand du centre vers les extrémités de cette fibre  
et aux fibres adjacentes. Cette propagation est un phénomène  
important car elle permet la contraction successive de toutes les  
fibres du muscle.

La stimulation initiale du coeur s'opère à l'extrémité  
supérieure de l'oreillette droite et se propage, à la manière  
d'une onde, tout autour du muscle cardiaque provoquant la  
contraction des oreillettes avant celle des ventricules.[17]

#### 3.3.2. LES QUATRES PHASES DU CYCLE CARDIAQUE

##### 1. Repos - remplissage :

Par suite du relâchement du muscle ventriculaire après  
son dernier battement, la pression du ventricule est tombée à  
un niveau très bas. Lorsqu'elle est inférieure à la pression  
règnant dans l'oreillette, la valve mitrale s'ouvre et le sang



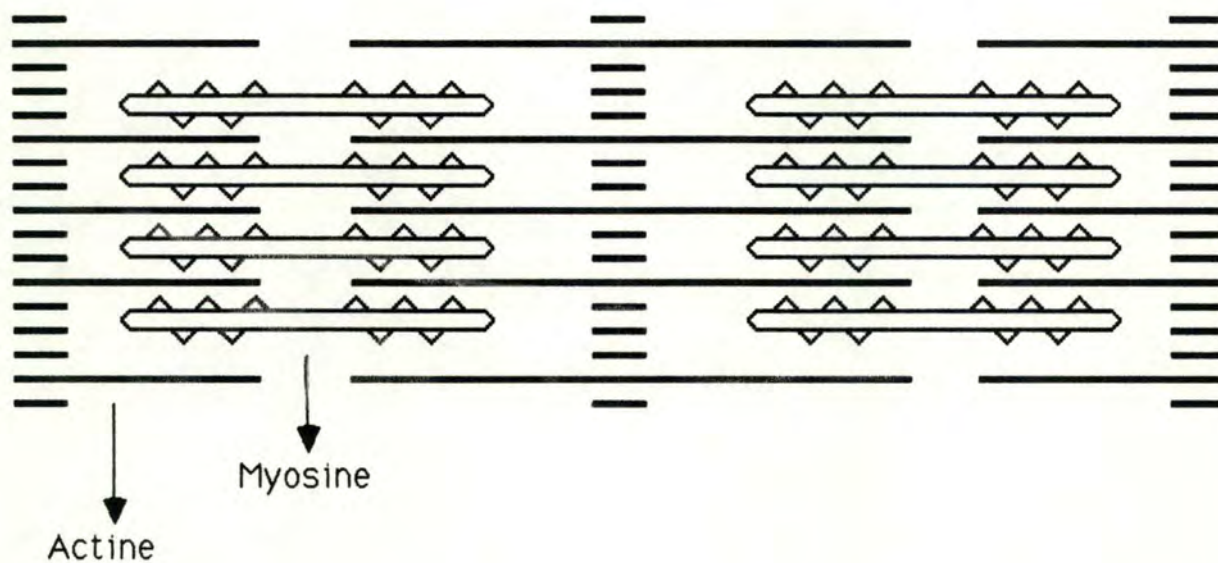


Figure 3.2 : Schéma de myofibrilles

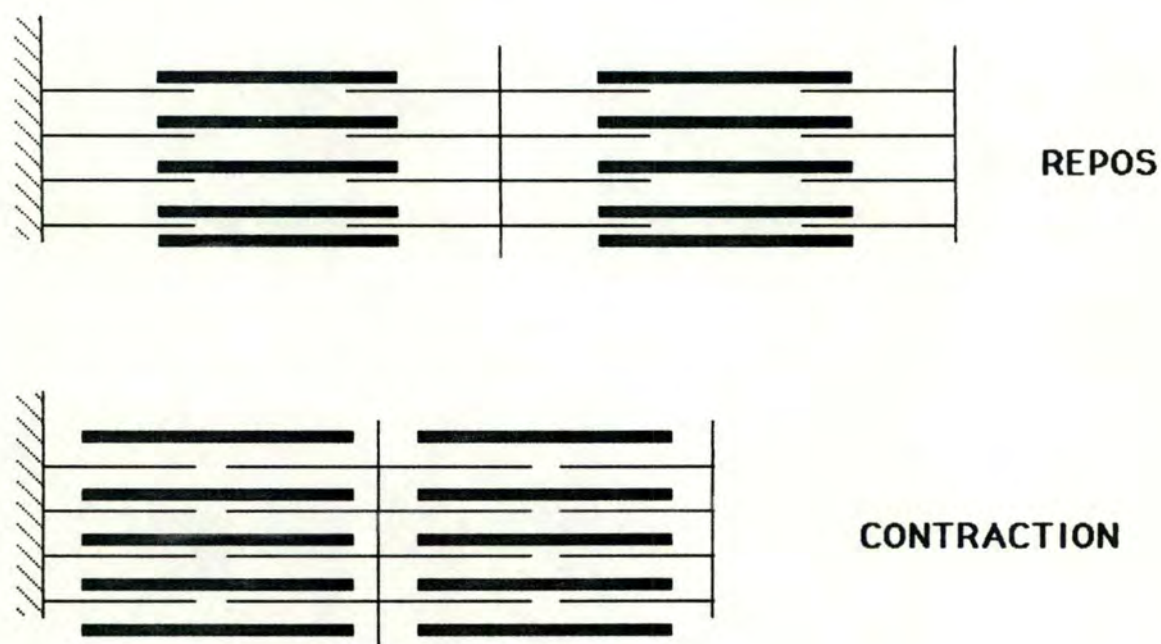


FIGURE 3.3 : Schéma des modifications des myofibrilles selon que le muscle est au repos ou contracté



s'écoule dans le ventricule. Le muscle ventriculaire se relâche davantage; ainsi la pression ventriculaire continue à tomber pendant un court instant et le remplissage devient rapide vu la différence de pression assez importante entre les deux chambres cardiaques.

Cette phase d'afflux rapide est suivie d'une autre moins rapide car la pression, conséquence du volume, a augmenté au sein du ventricule. A la fin de cette période de remplissage lent, a lieu le début de la contraction cardiaque. La stimulation électrique arrive au niveau de l'oreillette qui se contracte et la contraction de l'oreillette achève le remplissage du ventricule. Cette contraction entraîne une remontée passagère de la pression dans l'oreillette causant une reprise d'écoulement rapide et contribuant pour environ 20 à 30 % au remplissage du ventricule. Le volume de sang contenu dans celui-ci est alors maximal (volume télédiastolique).

Pendant cette phase, la valve d'échappement (aortique) est fermée car la pression de l'aorte est bien supérieure à celle du ventricule.

## 2. Contraction isovolumétrique

Le début de la contraction du muscle ventriculaire provoque la montée de sa pression et la valve mitrale se ferme. Le ventricule est maintenant un espace clos puisque les deux valvules sont fermées. Aussi longtemps qu'elles le restent, il ne peut y avoir de changement de volume d'où le nom donné à cette phase. La contraction, qui se poursuit, contribue à une augmentation très rapide de la pression du sang dans le ventricule.

Pendant le remplissage et la contraction isovolumétrique, la pression achève de baisser dans l'aorte car l'écoulement du sang de l'artère vers tout le corps (réseau périphérique ou systémique) se poursuit.

## 3. Contraction éjection

Au moment où la pression ventriculaire s'élève au-dessus de la pression restante dans l'aorte, la valve aortique s'ouvre. Le sang du ventricule est alors éjecté très rapidement dans l'aorte.

Pendant la première partie de l'éjection, la contraction qui se poursuit provoque une hausse de la pression ventriculaire et l'écoulement du sang est dû à la faible différence de pression entre le ventricule et l'aorte. Le volume de sang dans l'aorte augmente car elle se remplit plus rapidement qu'elle ne se vide. Sa pression augmente donc également et va



de pair avec celle du ventricule.

A la fin de l'éjection, la force de contraction diminue et le volume de sang contenu dans le ventricule est réduit. Ces deux effets conjugués causent une diminution de la pression ventriculaire. Parallèlement, l'aorte commence à se vider et sa pression diminue.

#### 4. Relaxation isovolumétrique

La contraction du ventricule cesse et est suivie d'un relâchement très rapide du muscle cardiaque. La pression ventriculaire tombe nettement en dessous de la pression aortique et la valve aortique se ferme. Le ventricule est à nouveau une chambre close dont le volume ne peut pas changer. La pression tombe rapidement et finalement jusqu'à un niveau inférieur à la pression auriculaire produisant l'ouverture de la valve mitrale et le cycle recommence.[18]

Le graphe de la figure 3.4 donne un aperçu de l'évolution de la pression du ventricule et de l'aorte au cours des 4 cycles et la figure 3.5 est le graphe de l'évolution du volume ventriculaire pendant ces cycles.

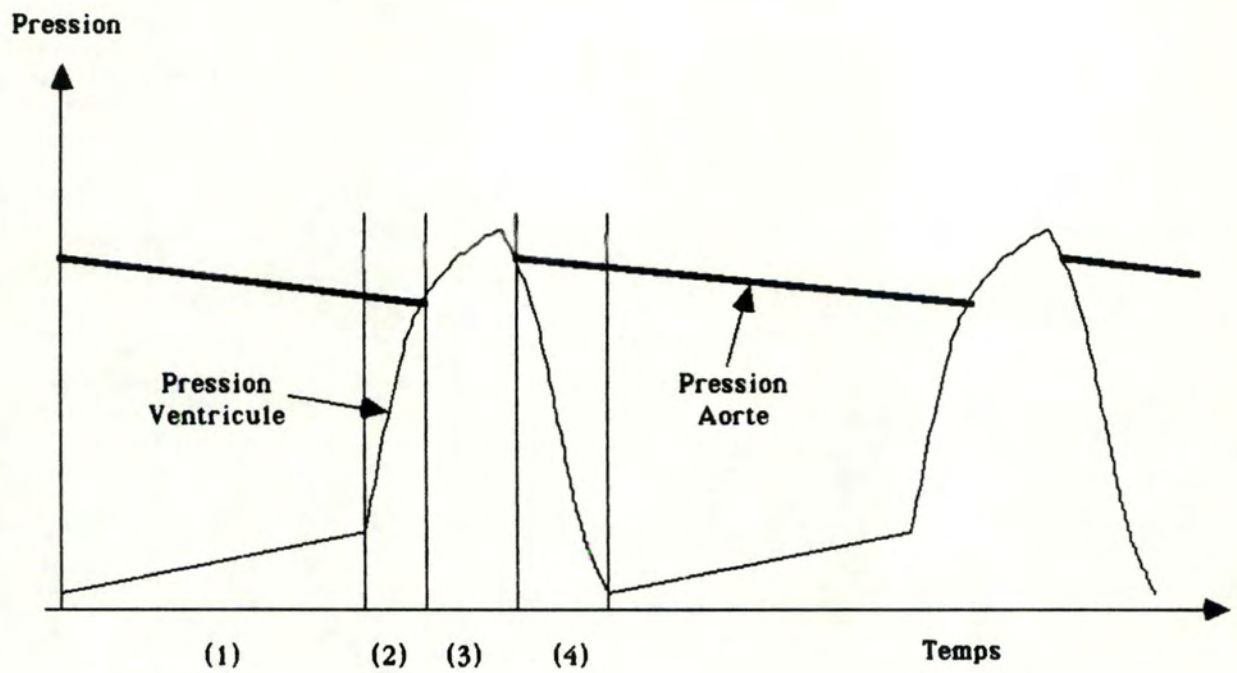
### 3.4. FACTEURS DE PERFORMANCE DU COEUR

---

Les facteurs de performance sont des coefficients qui permettent d'évaluer la qualité d'un coeur. Nous comptons parmi eux :

- la pression dans l'oreillette
- la pression dans l'aorte
- le rythme cardiaque
- la raideur du ventricule
- la qualité des valves

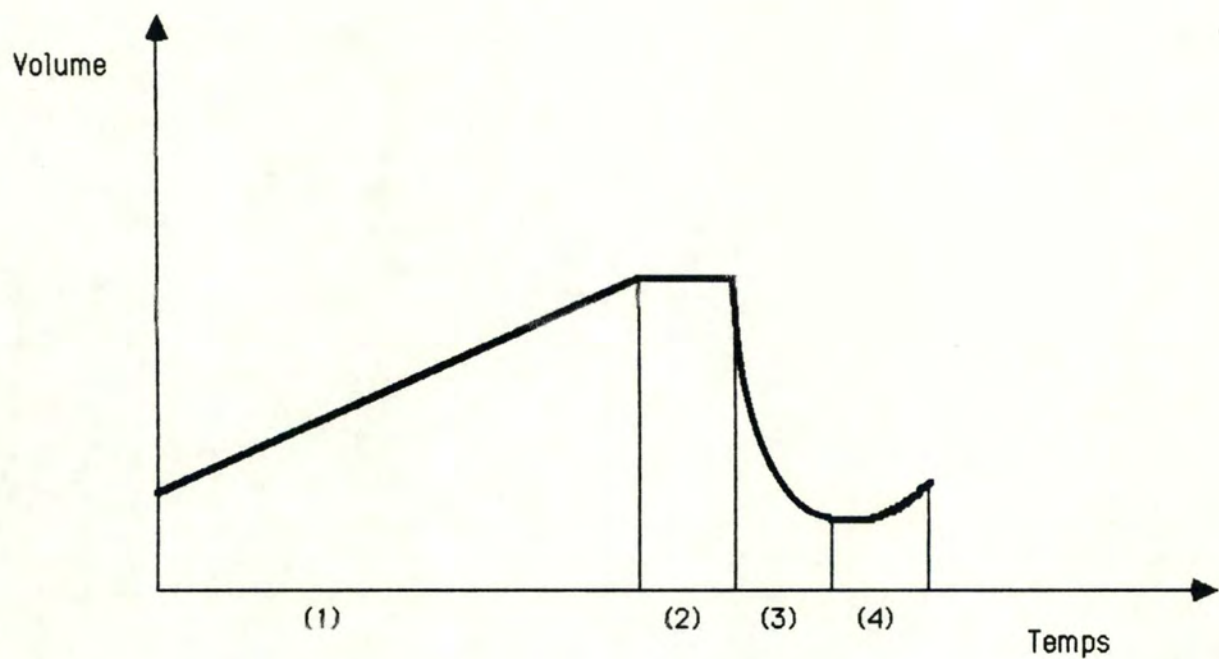
Parmi ces facteurs, certains déterminent la qualité du remplissage tandis que d'autres permettent une estimation de la performance du ventricule. Examinons les conséquences d'un coeur non performant.



- (1) : phase de repos
- (2) : phase de contraction isovolumétrique
- (3) : phase de contraction-éjection
- (4) : phase de relaxation

Figure 3.4. : Graphe pression-temps





(1) : Phase de repos

(2) : Phase de contraction isovolumétrique

(3) : Phase de contraction-éjection

(4) : Phase de relaxation

Figure 3.5 : Graphe Volume-Temps pour le ventricule gauche



### 3.4.1. FACTEURS DETERMINANT LE REMPLISSAGE DU VENTRICULE

#### 1. Pression de l'oreillette :

Si la pression auriculaire est trop faible, la valve mitrale se ferme avant que le volume de sang contenu dans le ventricule n'ait atteint son maximum.

Si la pression auriculaire est trop élevée, le volume de sang a tendance à dépasser ce maximum et à tendre fortement le muscle ventriculaire. La consommation d'énergie nécessaire à la tension des fibres du myocarde lors de la contraction sera trop importante.

#### 2. Rythme cardiaque (espace de temps entre 2 contractions)

Si le rythme est court, la contraction ventriculaire se produit rapidement provoquant une augmentation de pression et une fermeture prématurée de la valve mitrale. Le ventricule n'aura peut-être pas eu le temps de se remplir au maximum.

#### 3. Qualité de la valve mitrale

Si la valve mitrale, lorsqu'elle est ouverte, offre une résistance non négligeable à l'afflux de sang s'écoulant dans le ventricule, on dit qu'elle est sténosée. Cette sténose peut provoquer une diminution du flux sanguin et empêcher un remplissage maximal. [17]

### 3.4.2. FACTEURS DETERMINANT LA PERFORMANCE DU VENTRICULE

#### 1. Remplissage :

Un coeur trop rempli dépense une part trop importante de son énergie pour se contracter car les fibres du muscle sont très tendus . Il ne reste alors pas assez d'énergie pour augmenter suffisamment la pression ventriculaire.

Un coeur trop peu rempli ne peut approvisionner correctement le corps en sang oxygéné.

#### 2. Raideur du muscle :

Le muscle cardiaque lors de sa phase de remplissage doit se comporter tel un sac souple de telle sorte qu'une augmentation de volume ne provoque qu'une légère hausse de pression. Ainsi, la pression ventriculaire augmente lentement et n'égale celle de l'oreillette qu'une fois le ventricule



bien rempli.

Par contre la capacité de contraction des myofibrilles doit être suffisante pour compresser le ventricule et lui permettre d'éjecter dans l'aorte une quantité importante de sang.

### 3. Pression de l'aorte :

Si la pression aortique est trop élevée, la pression ventriculaire risque, lors de la contraction, de ne pas atteindre celle de l'aorte. Ainsi le ventricule ne peut pas éjecter et, par conséquent, alimenter en sang les vaisseaux capillaires du corps.[17]

## 3.5 PATHOLOGIES CARACTERISTIQUES

---

Nous présenterons dans ce paragraphe les anomalies cardiaques les plus courantes et leurs conséquences néfastes sur l'organisme. Ces pathologies sont l'artériosclérose, la sténose et l'insuffisance des valves.

### 3.5.1. ARTERIOSCLEROSE

La graisse ingurgitée en excès se dépose à l'intérieur des artères et se calcifie. La paroi artérielle en devient de moins en moins souple et perd sa capacité de dilatation. Le diamètre interne de l'artère diminue progressivement et le volume de sang pouvant passer se réduit. Ce phénomène est appelé l'artériosclérose.

Le développement de ce phénomène se produit généralement à l'entrée des artères coronaires. Les coronaires sont les vaisseaux sanguins qui alimentent le coeur en sang oxygéné.

La première conséquence de l'artériosclérose est la suivante. Si le diamètre des artères coronaires diminue, l'alimentation en sang devient insuffisante et certaines parties du muscle cardiaque ne parviennent plus à fonctionner correctement. La force de pompage en est gravement réduite. Inversement, le



besoin en oxygène augmente car le coeur doit compenser ces dysfonctionnements et réussir, en dépit de ceux-ci, une bonne contraction. Par conséquent, les cellules vaillantes volent l'oxygène de leurs voisines qui, à leur tour, ne peuvent remplir correctement leur fonction. Progressivement le nombre de cellules mortes se multiplie et le coeur ne parvient plus à battre. C'est l'infarctus.

Un deuxième phénomène consécutif à l'artériosclérose pouvant conduire à l'infarctus est l'occlusion d'une artère coronaire. S'il y a occlusion, le flux sanguin cesse au delà de l'obstruction. La surface du muscle cardiaque qui n'a pas reçu de sang en suffisance ne peut plus remplir sa fonction. Le coeur continue d'éjecter et le sang s'accumule dans l'artère bouchée et stagne. Cette stagnation provoque une dilatation du vaisseau. Parallèlement, les fibres musculaires utilisent les derniers vestiges d'oxygène qui se raréfie de plus en plus. Certaines cellules du coeur commencent à mourir et le phénomène se termine comme dans le cas précédant.

### 3.5.2. STENOSE ET INSUFFISANCE AORTIQUE

La valve aortique est dite sténosée si elle permet une ouverture valvulaire trop petite pour une éjection aisée du sang provenant du ventricule gauche.

La valve aortique est insuffisante si elle ne peut se fermer complètement et ne peut empêcher le reflux de sang de l'aorte vers le ventricule gauche durant la diastole.

Dans le cas d'une sténose aortique, le ventricule gauche ne peut se vider convenablement, tandis que lors d'une insuffisance aortique, le sang retourne vers le ventricule après que celui-ci se soit vidé. Le volume net éjecté en est alors réduit.

### 3. 5. 3. STENOSE ET INSUFFISANCE MITRALE

La valve mitrale est sténosée si elle ne peut s'ouvrir complètement lors du remplissage ventriculaire. La quantité de sang coulant de l'oreillette gauche vers le ventricule gauche en est réduit.

Si la valve mitrale est insuffisante, elle ne peut se fermer correctement et une partie importante du sang qui a coulé vers le ventricule gauche revient dans l'oreillette gauche pendant la systole plutôt que d'être pompée dans l'aorte. Dans ce cas, le volume net de sang du ventricule est inférieur à la normale. [17]

Les principaux principes régissant le fonctionnement du coeur qui vous ont été présentés dans ce chapitre constituent une source importante de connaissances pour la construction du système à venir.



## CHAPITRE 4

### MODELES VENTRICULAIRES

#### 4.1 INTRODUCTION

---

La modélisation d'un objet réel suppose un compromis entre la simplicité et la puissance explicative du modèle. La simplicité est caractérisée par le genre du modèle et le nombre de paramètres pris en considération. La puissance explicative est mesurée par la part du comportement du système réel qui peut être reproduite par ce modèle.

Un modèle minimal est un modèle satisfaisant au mieux au compromis entre ces deux concepts. Il est le plus simple possible et possède une puissance explicative suffisante pour mener à bien l'étude désirée. Un modèle minimal doit :

- être une abstraction du système physique qui retient les paramètres nécessaires à une bonne étude,
- prendre en compte uniquement les paramètres qui peuvent être estimés (identifiés) avec une précision raisonnable. Ces paramètres doivent être en correspondance avec ceux du système physique,
- présenter un comportement similaire à celui du système physique, (La similitude est un concept imprécis. Quand peut-on dire qu'un modèle est similaire à un système physique ?)
- être le modèle le plus simple qui satisfait aux exigences précitées.

Il est reconnu que les modèles hydrauliques et électriques présentés ci-après sont des modèles minimaux du ventricule gauche. [20]



## 4.2 MODELE HYDRAULIQUE

---

Le système cardiaque tel qu'il a été présenté précédemment peut être représenté par un modèle hydraulique. Par rapport au système réel, ce modèle suppose les simplifications suivantes:

- 1) Dans le système réel, la pression de l'oreillette augmente lors de sa contraction. Le remplissage ne s'effectue donc pas à pression constante. Dans le modèle, l'oreillette gauche n'est pas représentée. Le système de remplissage du ventricule est assimilé à un réservoir de liquide suffisamment grand pour qu'une variation de volume n'entraîne qu'une modification négligeable de la pression du réservoir. [19, p.145 fig 4]
- 2) Le ventricule au repos est représenté par un sac souple de manière à ce qu'une augmentation de volume ne provoque qu'une faible poussée de pression.
- 3) L'aorte est également représentée par un sac dont la souplesse est moindre par rapport à celle du sac ventriculaire au repos. La variation de la souplesse aortique durant le battement cardiaque n'est pas considérée.
- 4) Le réservoir est relié au sac ventriculaire par un tuyau contenant une valve qui s'ouvre si la pression du réservoir est supérieure à la pression du ventricule. Le sac ventriculaire communique avec le sac aortique par la même sorte de tuyau.
- 5) Le sac aortique a la possibilité de se vider par un troisième conduit de circonférence plus petite représentant le réseau systémique et sa résistance élevée au passage du flux sanguin.
- 6) La contraction du ventricule est assimilée à une pression extérieure exercée sur le sac ventriculaire. Les phénomènes de contraction, relaxation, repos, propres au muscle cardiaque peuvent faire l'objet d'une métaphore :  
Imaginons une boîte contenant des élastiques. Le nombre d'élastiques est représentatif de la raideur maximale que le ventricule peut atteindre au plus fort de sa contraction. Si la raideur maximale est élevée, le nombre d'élastiques sera grand. Pendant la phase de repos, le contenu du sac ventricu-



laire est soumis à une faible pression. Cette pression est, en fait, conséquence de la raideur initiale du sac. La contraction est déclenchée par une mise successive, autour du sac ventriculaire, des élastiques de la boîte. Ceci a pour conséquence d'augmenter progressivement la raideur du sac ventriculaire. Une fois que tous les élastiques sont mis en place, on les retire un à un, symbolisant ainsi la relaxation; ce jeu se répétant pour les cycles suivants.

Ce modèle préssuppose donc les hypothèses suivantes :

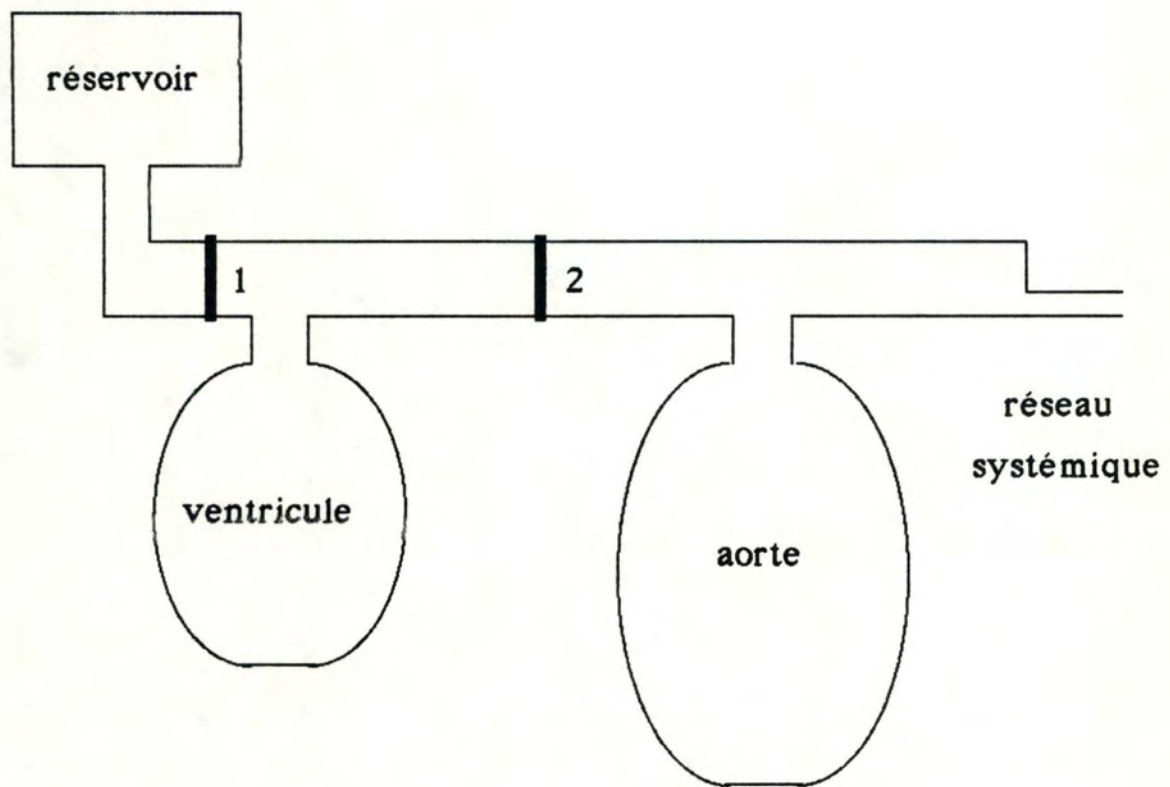
- 1) Le ventricule est alimenté par un réservoir ouvert à pression constante.
- 2) Le ventricule est un sac souple dont il est possible d'augmenter la raideur en plaçant des élastiques autour de lui.
- 3) L'aorte est un sac moins souple que celui du ventricule au repos.
- 4) Les valves mitrale et aortique sont représentées par des valves s'ouvrant quand la pression en aval est supérieure à la pression en amont.
- 5) Le liquide s'écoulant à travers le tuyau systémique n'alimente pas le réservoir contrairement au sang qui circule en boucle fermée. Les phénomènes liés à l'ouverture et fermeture des valves (onde dicrote, onde C, ...) ne sont pas considérés.
- 6) On suppose l'écoulement laminaire de sang.

[17]

Trois grandeurs déterminent à elles seules l'évolution temporelle de chaque élément :

- le débit :  $F$
- la pression locale :  $P$
- le volume de sang confiné dans une partie du réseau :  $V$

La représentation graphique de ce modèle est présentée à la figure 4.1.



1 : valve mitrale  
2 : valve aortique

Figure 4.1 : Schéma du modèle hydraulique



## 4. 3 MODELE ELECTRIQUE

---

### 4. 3. 1 PRESENTATION DU MODELE

La figure 4.2 est le dessin du circuit électrique représentant un ventricule gauche isolé recevant du sang d'une source de pression constante ( $E$ ) et délivrant du sang à un réseau simulant la charge présentée au ventricule par l'aorte et le réseau systémique. Cette charge est communément appelée "after-load" ou "postcharge".

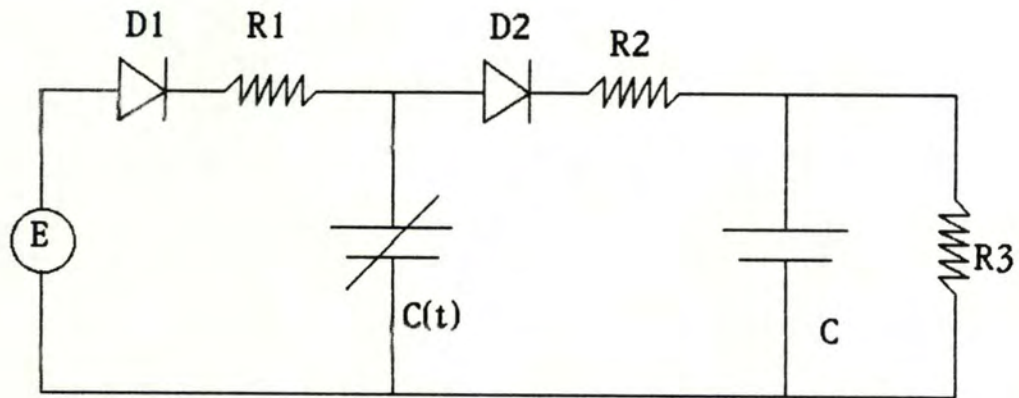
Les propriétés de contractilité du ventricule sont modélisées par un capaciteur à capacité variable ( $C(t)$ ). Un capaciteur est un accumulateur d'énergie électrique et la capacité, ou capacitance, correspond à la valeur de la compliance (souplesse) du ventricule modélisé dans le domaine hydraulique. Le ventricule à l'état relaxé est représenté par une capacitance élevée tandis qu'à l'état contracté la capacitance est faible. Le capaciteur est couramment utilisé pour représenter des éléments cardiaques sujets à un remplissage (ventricule, veine, artère). La compliance varie en fonction du temps et la fonction  $C(t)$  est supposée connue expérimentalement.

Les valves mitrale et aortique sont représentées par des diodes ( $D1$  et  $D2$ ), présentant une résistance nulle dans un sens et infinie dans l'autre. Elles sont mises en série avec une résistance qui incorpore les pertes de charge dues au frottement dans les valves ( $R1$  et  $R2$ ).

La source de tension ( $E$ ) est l'équivalent du réservoir dans le modèle hydraulique.

$R2$ ,  $C$ ,  $R3$  symbolisent la postcharge ventriculaire. [8]

Ici également, ce modèle est une simplification de la réalité et beaucoup de phénomènes existants ne sont pas pris en compte.



**E : Source de tension**

**D1 : Diode représentant la valve mitrale**

**D2 : Diode représentant la valve aortique**

**C (t) : Capaciteur représentant le ventricule gauche**

**R2, C, R3 : Postcharge ventriculaire**

**Figure 4.2 : Schéma du modèle électrique**



Tout comme pour le modèle hydraulique, trois grandeurs localisées suffisent, ici aussi, à déterminer les différents éléments de ce modèle :

- la tension : E
- le courant : I
- la charge : Q

Il a été établi qu'il existe une analogie entre systèmes hydrauliques et électriques. En effet, les 3 grandeurs caractérisant l'évolution de chaque élément du modèle hydraulique et celles du modèle électrique ont une signification analogue :

Tension E = Pression P

Courant I = Débit F

Charge Q = Volume V

De même, les lois hydrauliques sont équivalentes aux lois électriques :

Exemple

$$E = R * I$$

$$P = R * F \text{ où } R \text{ est une résistance}$$

[21]

Par conséquent, nous avons la possibilité de considérer l'un ou l'autre des modèles puisqu'ils représentent de manière analogue le même objet du réel.

#### 4.3.2 EQUATIONS TIREES DU MODELE

Soit E : source de tension idéalement constante et indépendante de tout autre élément du circuit

$C_v(t)$  : compliance de C(t) à l'instant t

$C_a$  : compliance de C (constante)

$Q[CV]$  : charge de C(t)

$Q[CA]$  : charge de C

(cfr. fig. 4.2)

La définition de la capacité étant le rapport de la charge et de la tension, il est établi que :

1) Si  $E[CV]$  est la tension de  $C(t)$

$$\text{Alors } E[CV] = \frac{Q[CV]}{Cv(t)}$$

2) Si  $E[CA]$  est la tension de  $C$

$$\text{Alors } E[CA] = \frac{Q[CA]}{Ca}$$

3) Si  $E > E[CV]$  alors  $I_{D1} = \frac{E - E[CV]}{R1_{D1} \text{ passante}}$

$$\text{sinon } I_{D1} = \frac{E - E[CV]}{R1_{D1} \text{ bloquante}}$$

4) Si  $E[CV] > E[CA]$  alors  $I_{D2} = \frac{E[CV] - E[CA]}{R2_{D2} \text{ passante}}$

$$\text{sinon } I_{D2} = \frac{E[CV] - E[CA]}{R2_{D2} \text{ bloquante}}$$

Par analogie et par extension :

Soit  $P_v$  : pression du ventricule

$P_a$  : pression de l'aorte

$P_{res}$  : pression du réservoir

$P_{sys}$  : pression du réseau systémique

débit mitrale : flux passant par la valve mitrale

débit aortique : flux passant par la valve aortique

débit per : Flux passant par le réseau systémique

$vol_v$  : volume dans le ventricule

$vol_a$  : volume dans l'aorte

$R_{mitrale}$  : résistance de la valve mitrale

$R_{aortique}$  : résistance de la valve aortique

$R_{per}$  : résistance du réseau systémique

$C(t)$  : élasticité (compliance) du ventricule à l'instant  $t$



$$C(t) = \frac{1}{\text{raideur ventricule } (t)}$$

Nv : nombre d'élastiques autour du ventricule à l'instant t = raideur ventricule (t)

Ca : élasticité (compliance) de l'aorte

$$Ca = \frac{1}{\text{raideur de l'aorte}}$$

Volv et Volao sont des valeurs connues.

Rmitrale, Raortique, Rper, Pres et Psys sont des valeurs données.

Volv

$$1) P_v = \frac{\text{Volv}}{C(t)} = \text{Volv} * \text{raideur ventricule } (t) = \text{Volv} * N_v$$

Volao

$$2) P_a = \frac{\text{Volao}}{Ca} = \text{Volao} * \text{raideur aorte}$$

$$3) \text{débit mitrale} = \frac{\text{Pres} - P_v}{R_{\text{mitrale}}}$$

$$4) \text{débit aortique} = \frac{P_v - P_a}{R_{\text{aortique}}}$$

$$5) \text{débit per} = \frac{P_a - P_{\text{sys}}}{R_{\text{per}}}$$

$$6) \text{variation volume ventricule} = \text{débit mitrale} - \text{débit aortique}$$

$$7) \text{variation volume aorte} = \text{débit aortique} - \text{débit per}$$

Ces équations sont des connaissances primordiales à la constitution de notre système. Elles sont utilisées pour calculer les valeurs caractéristiques du coeur gauche et, dès lors, font partie de l'ensemble des connaissances exposé dans le chapitre suivant.

## CHAPITRE 5

### ACQUISITION DES CONNAISSANCES

#### 5.1 INTRODUCTION

---

L'élaboration d'une base de connaissances s'appuie sur un dialogue entre un expert travaillant dans le domaine concerné et le concepteur de cette base. Pour le cas qui nous préoccupe, l'expert est le professeur de physiologie et nous sommes les concepteurs.

Ce dialogue entre expert et concepteur permet le transfert des connaissances nécessaires à la construction du système. Dans un premier temps, l'expert tente d'expliquer la manière dont il raisonne sur un problème particulier tandis que le concepteur écoute et essaie de saisir les méthodes de résolution du problème existant. Dans un second temps, le concepteur crée une modélisation de ce raisonnement et réalise un prototype qui génère des comportements de résolution que l'expert peut critiquer et améliorer. [23] La formulation du raisonnement peut être effectuée en termes de règles reproduisant chaque étape de pensée.

Les règles qui vous seront présentées sont tirées des équations régissant le modèle électrique du coeur présenté ci-avant et du raisonnement de l'expert interprétant les résultats donnés par ce modèle. Avant de détailler ces règles, il est important de définir les différents paramètres et variables pour cerner davantage leur rôle au sein du système.



## 5.2 VARIABLES ET PARAMETRES DU SYSTEME

---

### 5.2.1. LES VARIABLES

Les variables décrivent l'état du coeur à un instant donné.

#### VARIABLES D'ETAT :

Les variables d'état sont des variables à partir desquelles il est possible de calculer la valeur de toute autre variable.

1. Volume de sang dans le ventricule à un instant donné ( $V_{olv}$ )  
ou  
Pression dans le ventricule à un instant donné ( $P_v$ )
2. Volume de sang dans l'aorte à un instant donné ( $V_{olao}$ )  
ou  
Pression dans l'aorte à un instant donné ( $P_a$ )

Le système est dit déterminé s'il est possible, à partir de la valeur des variables d'état, de calculer la valeur des autres variables. Si les valeurs de  $V_{olv}$  ou  $P_v$  et  $V_{olao}$  ou  $P_a$  sont connues, le système cardiaque est déterminé.

#### VARIABLE D'ENTREE :

Une variable d'entrée représente une donnée extérieure au système ayant une influence sur la valeur des variables de ce système.

- Raideur du ventricule à un instant donné (= nombre d'élastiques autour du coeur à un instant donné,  $N_v$ )

#### AUTRES VARIABLES

1. Débit passant par la valve mitrale (débit-mit)
2. Débit passant par la valve aortique (débit-aortique)
3. Débit vers le réseau périphérique (débit-per)

4. Variation du volume de sang dans le ventricule (varvolv)
5. Variation du volume de sang dans l'aorte (varvolao)

### 5.2.2. LES PARAMETRES

Les paramètres du système représentent des éléments physiques du coeur et influencent son comportement. Cette influence est prise en compte lors du calcul des valeurs de variables. Ils sont indispensables à ces calculs.

Les valeurs des paramètres restent constantes pendant toute la durée de l'exécution d'une session mais peuvent être modifiées entre chaque session.

1. Pression du réservoir (Pres)
2. Résistance passante de la valve aortique (Raorte-pas)
3. Résistance bloquante de la valve aortique (Raorte-bloc)
4. Résistance passante de la valve mitrale (Rmitrale-pas)
5. Résistance bloquante de la valve mitrale (Rmitrale-bloc)
6. Résistance du réseau périphérique (Rper)
7. Pression du réseau périphérique (Psys)
8. Raideur de l'aorte (raid-a)
9. Rythme cardiaque (Rythme)
10. Elastance (raideur) maximale du ventricule (Emax)

## 5.3 REGLES DE RAISONNEMENT DETERMINANT L'ETAT DU SYSTEME

---

### 5.3.1. CALCUL DES PRESSIONS

La pression dans un élément du coeur est fonction du volume et de la raideur de cet élément.

- pression du ventricule = volume ventricule \* raideur du ventricule
- pression de l'aorte = volume aorte \* raideur aorte



### 5.3.2. CALCUL DES VOLUMES

Le volume de sang d'un élément cardiaque est fonction de la pression et de la raideur de cet élément.

$$\begin{array}{l} \text{pression ventricule} \\ - \text{ volume ventricule} = \frac{\text{-----}}{\text{raideur ventricule}} \end{array}$$

$$\begin{array}{l} \text{pression aorte} \\ - \text{ volume aorte} = \frac{\text{-----}}{\text{raideur aorte}} \end{array}$$

### 5.3.3. DETERMINATION DE LA POSITION DES VALVES

Une valve est ouverte si la pression en amont est plus grande que la pression en aval et fermée dans le cas contraire.

- SI pression réservoir  $\geq$  pression ventricule  
ALORS valve mitrale ouverte
- SI pression réservoir  $<$  pression ventricule  
ALORS valve mitrale fermée
- SI pression ventricule  $\geq$  pression aorte  
ALORS valve aortique ouverte
- SI pression ventricule  $<$  pression aorte  
ALORS valve mitrale fermée

### 5.3.4. CALCUL DES DEBITS

Le débit d'une valve est fonction de la résistance offerte par cette valve et des pressions en aval et en amont, le débit est positif si la valve est ouverte et négatif ou nul si la valve est fermée.

- SI valve mitrale ouverte

ALORS débit valve mitrale > 0

et

$$\text{débit valve mitrale} = \frac{\text{Pression réservoir} - \text{Pression ventr.}}{\text{Résistance passante valve mitrale}}$$

- SI valve aortique ouverte

ALORS débit valve aortique > 0

et

$$\text{débit valve aortique} = \frac{\text{Pression ventr.} - \text{Pression aorte}}{\text{Résistance passante valve aortique}}$$

Pour un coeur physiologiquement normal, les valves une fois ouvertes offrent une résistance négligeable et le flux passant par les valves est égal à la différence des pressions.

Si la valve est sténosée (elle ne s'ouvre pas complètement), elle offre une résistance supérieure à la normale et non négligeable. Le débit en est réduit.

- SI valve mitrale fermée

ALORS débit valve mitrale <= 0

et

$$\text{débit valve mitrale} = \frac{\text{Pression réservoir} - \text{Pression ventr.}}{\text{Résistance bloquante valve mitrale}}$$

- SI valve aortique fermée

ALORS débit valve aortique <= 0

et

$$\text{débit valve aortique} = \frac{\text{Pression ventr.} - \text{Pression aorte}}{\text{Résistance bloquante valve aortique}}$$

Pour un coeur physiologiquement normal, les valves une fois fermées offrent une résistance très élevée de manière à éviter tout reflux du sang. Cette résistance bloquante est considérée comme ayant une valeur infinie et le débit est nul.

Si la valve est insuffisante (elle n'est pas complètement fermée), elle offre une résistance inférieure et le débit est négatif.



$$\begin{array}{rcl} & \text{Pres. aorte - Pres. périph.} & \\ - \text{débit vers le réseau périphérique} = & \frac{\text{-----}}{\text{Résistance périphérique}} & \end{array}$$

La pression phériphérique est négligeable et est considérée comme nulle.

### 5.3.5. CALCUL DE LA VARIATIONS DES VOLUMES

La variation d'un volume est fonction du débit entrant et du débit sortant de l'élément cardiaque.

- Variation volume ventricule pendant un intervalle de temps T  
= Volume entrant pendant T - Volume sortant pendant T

où Volume entrant pendant T = débit valve mitrale \* durée de  
l'intervalle T

Volume sortant pendant T = débit valve aortique \* durée de  
l'intervalle T

- Variation volume aorte pendant un intervalle de temps T  
= volume entrant pendant T - volume sortant pendant T

où Volume entrant pendant T = débit valve aortique pendant T  
Volume sortant pendant T = débit réseau périphérique pendant T

## 5.4 DESCRIPTION GENERALE DES PHASES DU CYCLE CARDIAQUE VIA LES VARIABLES DU SYSTEME

---

### 5.4.1. PHASE DE REPOS

- SI le nombre d'élastiques autour du ventricule = nombre résiduel d'élastiques  
ALORS le coeur est dans sa phase de repos
- SI le coeur est dans sa phase de repos  
ALORS 1. la pression du ventricule est très inférieure à la pression de l'aorte  
2. la pression du ventricule augmente peu et lentement  
3. le volume dans le ventricule est compris entre le volume incompressible et le volume maximal du ventricule  
4. le volume dans le ventricule augmente beaucoup et lentement  
5. la pression de l'aorte diminue peu et lentement

### 5.4.2. PHASE DE CONTRACTION ISOVOLUMETRIQUE

- SI le nombre d'élastiques autour du ventricule croît  
et la valve aortique est fermée  
et la valve mitrale est fermée  
ALORS le coeur est dans sa phase de contraction isovolumétrique
- SI le coeur est dans sa phase de contraction-isovolumétrique  
ALORS 1. la pression dans le ventricule est supérieure à la pression dans le réservoir  
2. la pression dans le ventricule est inférieure à la pression dans l'aorte  
3. le volume dans le ventricule est plus ou moins égal au volume maximal  
4. le volume dans le ventricule est constant  
5. la pression dans le ventricule augmente beaucoup et rapidement  
6. le volume dans l'aorte diminue peu et lentement  
7. la pression dans l'aorte diminue peu et lentement



#### 5.4.3. PHASE DE CONTRACTION-EJECTION

- SI le nombre d'élastiques autour du ventricule croît  
et la valve aortique est ouverte  
ALORS le coeur est dans sa phase de contraction-éjection
- SI le coeur est dans sa phase de contraction-éjection  
ALORS 1. la pression dans le ventricule est légèrement  
supérieure à la pression dans l'aorte  
2. la pression dans le ventricule est fortement supérieure  
à la pression dans le réservoir  
3. le volume dans le ventricule diminue très vite au  
début et plus lentement à la fin  
4. la pression de l'aorte est fonction croissante de la  
pression du ventricule

REMARQUE : Fonction croissante :

- si la pression dans le ventricule augmente  
alors la pression dans l'aorte augmente
- si la pression dans le ventricule diminue  
alors la pression dans l'aorte diminue

#### 5.4.4. PHASE DE RELAXATION

- SI le nombre d'élastiques autour du coeur diminue  
ALORS le coeur est dans sa phase de relaxation
- SI le coeur est dans sa phase de relaxation  
ALORS 1. la pression dans le ventricule diminue rapidement  
et fortement  
2. la pression dans le ventricule est inférieure à la  
pression dans l'aorte  
3. au début de la phase la pression dans le ventricule  
est supérieure à la pression dans le réservoir et  
devient inférieure à la fin de la phase  
4. la pression dans l'aorte diminue peu et lentement  
5. le volume dans l'aorte diminue peu et lentement  
6. le volume dans le ventricule est constant tant que sa  
pression est supérieure à celle dans le réservoir et  
augmente très légèrement quand elle devient inférieure

L'ensemble des règles qui vous ont été présentées constitue, dans la bases de connaissances de notre système, la partie consacrée exclusivement aux connaissances physiologiques. A celles-ci doivent s'ajouter des connaissances relatives aux mécanismes de manipulation de grandeurs qualitatives exposés au chapitre suivant.



## CHAPITRE 6

### RAISONNEMENT QUALITATIF

#### 6.1 INTRODUCTION

---

Beaucoup de mots et de phrases dans le langage écrit et parlé sont vagues. Nous parlons d'une bière "froide", d'un taux d'intérêt "élevé", d'un collègue "intelligent" mais nous ne désirons pas spécifier la température à laquelle la bière cesse d'être froide, le pourcentage auquel un taux d'intérêt cesse d'être élevé ou le comportement qui qualifie un collègue d'intelligent. Intuitivement nous savons tous à quoi ressemble une bière froide et quel goût elle a, mais il n'existe pas de test de sensibilité capable de nous dire exactement quand une bière passe de l'état froid à l'état chaud. Peut-on affirmer que, si 4°C est considéré comme une température froide, 5°C n'est plus une température froide ? Quand on laisse un verre de bière glacé au soleil, il devient graduellement chaud. Il n'y a pas de critère précis disant quand une bière cesse d'appartenir à la classe des objets froids.

L'utilisation de mots (ou termes qualitatifs) dans la caractérisation des valeurs de variables implique un degré plus bas de précision par rapport à l'utilisation de nombres et le raisonnement effectué à partir d'un ensemble de propositions qualitatives (raisonnement qualitatif) est un raisonnement approximatif. Des conclusions imprécises et parfois incertaines sont déduites d'hypothèses imprécises. [1]

#### Exemple :

1. En termes qualitatifs, soient 2 hypothèses :

- X est vieux
- Y n'est plus très jeune

Nous pouvons en déduire une troisième proposition qualitative qui dirait que X est plus âgé que Y. Cette proposition n'est pas certaine puisque nous ne connaissons pas exactement l'âge de X et l'âge de Y.



2. En termes quantitatifs, soient 2 hypothèses :

- âge de X = 80 ans
- âge de Y = 50 ans

Nous déduirons avec certitude : âge de X > âge de Y

Le langage utilisé pour exprimer le comportement de différentes fonctions du corps humain est souvent qualitatif. En effet, ce comportement est décrit et expliqué par des termes tels que PETIT, GRAND, RAPIDEMENT,... L'usage de chiffres pour quantifier les valeurs des éléments d'un système est plutôt rare. Les médecins et les professeurs de physiologie raisonnent en termes de variations des valeurs par rapport aux valeurs physiologiquement normales, il leur suffit de savoir si une variable est plus petite ou plus grande que la normale. Un cardiologue dira d'un coeur qu'il bat "vite" ou plus vite que la normale, que la pression exercée par le sang est "faible" ou "forte" mais il ne dira pas que la pression artérielle est de 120 millimètres de mercure ou que le volume du sang est de 6,09 litres.

Puisque le but de ce travail est de reproduire le raisonnement d'un professeur qui explique l'évolution dans le temps du comportement du coeur gauche, le système créé à cette fin manipule des variables à valeurs qualitatives. Ces valeurs sont obtenues à partir d'équations manipulant des opérations à arguments qualitatifs. L'analyse de l'évolution d'un système se déroule en deux étapes. La première est la détermination de l'état du système en fonction des valeurs des ses variables et la deuxième est la propagation dans le temps de la valeur de ces variables pour étudier son état à l'instant suivant.

Un aperçu des différentes recherches déjà effectuées dans ce domaine est donné au paragraphe 6.2, le paragraphe 6.3 expose la solution retenue pour résoudre le problème du calcul qualitatif des valeurs des variables à l'instant courant et le paragraphe 6.4 énonce les principes retenus pour la propagation dans le temps des valeurs de ces variables.



Suite aux nombreuses lectures que nous avons faites afin d'en apprendre un peu plus sur l'état de la recherche dans le domaine du raisonnement qualitatif, nous pouvons distinguer, en quelque sorte, deux orientations différentes selon le type de systèmes à étudier. Certaines techniques traitent exclusivement des systèmes physiques [10] [11] [12] [26] [27], d'autres envisagent le raisonnement qualitatif imprécis sur des systèmes beaucoup plus généraux [1] [13] [28] [29] [30] [31]. Nous tentons ci-dessous de présenter la philosophie générale des méthodes qui nous ont paru les plus intéressantes.

Récemment, un certain nombre d'études ont été entreprises pour automatiser des techniques particulières de raisonnement qualitatif pour l'analyse de systèmes physiques. Une physique qualitative prédit et explique le comportement de mécanismes en termes qualitatifs. Les buts de cette physique sont 1) être beaucoup plus simple que les physiques classiques et pourtant retenir tous les éléments importants sans pour cela rentrer dans les détails mathématiques des variations continues des variables et des équations différentielles; 2) produire des explications causales faciles à comprendre sur un mécanisme physique; 3) fournir une base aux modèles de "commonsense" utiles aux systèmes experts à venir. [10]

Plusieurs théories ont déjà été élaborées à ce sujet. Retenons celles de DEKLEER et de FORBUS [11] [12]. Nous n'exposons pas l'intégralité de ces théories, nous en dégageons seulement les éléments qui nous semblent être les plus représentatifs pour le sujet qui nous préoccupe.

DEKLEER propose de définir le comportement d'un système à l'aide de confluences. Les confluences sont des équations différentielles qualitatives. Les variables de ces équations prennent des valeurs dans un espace de quantité défini par des intervalles ouverts sur la droite réelle. Il affirme que trois intervalles suffisent pour estimer la valeur des variables d'un système physique en fonction de ses confluences, pour déterminer l'état du système à un instant donné et prédire son état à l'instant suivant. Ces intervalles sont :  $<--$ ;  $0[$  ;  $[0$  ;  $]0$ ,  $-->$ . Par conséquent, les valeurs sont soit positives, soit nulles, soit négatives.



FORBUS, quant à lui, déduit l'état d'un système par l'établissement de relations d'ordre entre les différentes variables déterminant ainsi un espace de quantités.

#### Exemple

Si pression dans réservoir > pression dans ventricule  
Si pression dans ventricule < pression dans aorte

Il est possible de déduire de cet espace de quantité que le ventricule gauche se remplit de sang.

En ce qui concerne les techniques qui traitent du raisonnement qualitatif sur d'autres systèmes que les systèmes physiques, nous retenons principalement quatre approches [13].

La première est appelée l'approche engineering. Selon cette méthode le problème doit être formulé de manière à éviter une représentation explicite de l'incertitude. Il s'agit de poser des hypothèses sur le système afin de pouvoir raisonner sans tenir compte de l'incertitude initiale. Cependant, il faut veiller à ce que le degré d'abstraction par rapport au monde réel reste raisonnable. Dans le cas contraire, il est très probable que le nouveau système ne corresponde presque plus au problème original.

L'approche suivante est la méthode de diversification. La diversification semble être une stratégie appropriée pour lever l'incertitude de solutions alternatives pour lesquelles il existe un manque de preuve et plus aucune possibilité de preuve à venir. Elle consiste à envisager toutes les solutions possibles lorsqu'il y a manque de preuve de façon à s'assurer qu'au moins une soit valable.

La troisième méthode est celle qui est la plus souvent utilisée en intelligence artificielle pour traiter le problème du raisonnement qualitatif incertain : les inférences parallèles de certitude. D'une manière générale, cette technique éclate le raisonnement en deux processus plus ou moins dépendants. Suivant le premier processus, des conclusions sont dérivées comme si le raisonnement s'effectuait sur des données certaines. Le deuxième processus quant à lui, détermine dans quelle mesure les conclusions établies dans le premier sont fiables. La théorie de Bayes, celle des fonctions de croyance de Shafer-Dempster, la logique des ensembles flous de Zadeh et la théorie des Endorse-



ments de Rollinger sont à considérer dans ce troisième type de méthodes.

Enfin nous terminons par l'approche de contrôle. Contrairement aux trois autres méthodes, elle ne cherche pas à supprimer l'incertitude pour pouvoir raisonner sur des domaines certains. Elle est basée sur le fait que l'incertitude disparaît souvent lorsque le problème est résolu suivant une stratégie de contrôle conçue adéquatement. Le principe est qu'il faut tirer parti des traitements déjà effectués pour continuer la résolution du problème de manière efficace.

Si nous tentons de situer notre travail vis-à-vis de l'une ou l'autre de ces approches, nous remarquons qu'aucune ne peut satisfaire la réalisation de notre projet. Nous expliquons à présent notre position en quelques mots.

Les théories de physiques qualitatives semblent inadaptées au type de problème que nous devons résoudre. En effet, les plages de valeurs proposées par DEKLEER reflètent insuffisamment la pensée des spécialistes dans le domaine de la médecine. Il ne suffit pas de savoir si une valeur est positive pour donner une appréciation judicieuse de l'état du système. Il est nécessaire de connaître également l'ordre de grandeur de cette variable : est-elle proche de 0, ou non et, en d'autres termes, est-elle petite ou grande ? En outre, le système d'inégalités établi par FORBUS est intéressant mais, encore une fois, insuffisant car il ne donne aucun renseignement sur les valeurs des variables d'un mécanisme. Il nous permet de situer ces valeurs par rapport à d'autres mais ne donne pas la possibilité de quantifier ces valeurs.

Parmi les autres méthodes envisagées, nous laissons immédiatement de côté les approches de diversification et de contrôle. Dans la première le risque d'une explosion combinatoire n'est pas improbable puisque toutes les solutions possibles sont considérées dans la phase de détermination des opérations qualitatives. De plus, dans ce cas, aucune opération ne peut être réellement définie car à tout couple d'arguments correspondent certainement plusieurs résultats. La deuxième méthode ne nous intéresse absolument pas. Avoir déterminé l'état du système à un instant particulier ne nous donne aucune indication sur une résolution future possible du problème.

En ce qui concerne la technique des inférences de certi-



tude parallèles, nous signalons qu'initialement nous la considérons comme la plus satisfaisante. Cependant, lorsqu'il fut question d'accorder une probabilité ou un degré de croyance au résultat d'une opération, nous étions contraintes de le faire d'une manière tout à fait aléatoire. Cette démarche n'avait donc plus aucun sens.

Finalement, la technique à laquelle pourrait correspondre le plus notre démarche est l'approche d'engineering qui réduit l'incertitude par l'intermédiaire d'hypothèses sur le système. D'une manière générale nous pouvons dire que nous avons résolu le problème de la définition des opérations qualitatives en faisant référence au réel observé. A chaque couple d'arguments qualitatifs est associé un résultat qui tente de correspondre au mieux à la réalité du comportement du coeur gauche.

### 6.3 CALCUL DES VALEURS DES VARIABLES A L'INSTANT COURANT

---

La détermination de l'état d'un système à l'instant  $t$  peut être effectué en fonction de valeurs des variables à cet instant. Nous parlerons dorénavant des phases du système et non plus des états.

L'estimation des variables de notre système (point 5.2) est établie sur base de deux concepts:

1. les valeurs qualitatives que peuvent prendre les variables du système
2. les opérations qualitatives qui permettent de calculer les valeurs de toutes variables en fonction de la valeur des variables d'état

#### 6.3.1 CHOIX DES VALEURS QUALITATIVES

Selon la logique des prédicats, une proposition est soit vraie soit fausse. Il est possible de décider, sur base des règles et des faits contenus dans la base de connaissances, si une proposition est un théorème (est une proposition vraie), la



logique classique ne possédant que deux valeurs de vérité : vrai et faux (V, F).

Lorsqu'on traite des systèmes incluant un certain degré d'incertitude, il n'est pas toujours possible de décider si une proposition est entièrement vraie ou pas. Certains chercheurs en intelligence artificielle ont introduit des nuances supplémentaires à ces valeurs de vérité en ajoutant à l'ensemble [V, F] de départ des valeurs comme, par exemple, "peut-être vrai", "peut-être faux". [31][36]

Notre système possède les valeurs de vérité V et F mais nous pourrions également ajouter des termes à cet ensemble. Si un utilisateur veut connaître la pression du ventricule, il va poser au système la question suivante :  $is(pres-vent)$ . Le système va vérifier si oui ou non la pression peut être calculée et il va donner une valeur représentative de cette pression en fonction du contenu de la base de connaissances.

La réponse à une proposition n'est pas seulement la valeur V ou F mais est aussi une valeur qualitative donnant une approximation sur la grandeur de la variable au sujet de laquelle l'utilisateur a posé sa question. Ainsi nous pouvons considérer que les valeurs de vérité de notre système, outre les valeurs classiques V et F, sont définies par l'ensemble des valeurs qualitatives, que nous allons choisir, et qui représenteront les valeurs possibles de chaque variable du système.

## CONTEXTES

Les experts médicaux distinguent deux composants majeurs dans le système cardiaque : le ventricule gauche et l'aorte. Ceux-ci sont en relation uniquement au moment de l'éjection du sang dans l'artère et le comportement de l'un est alors influencé par le comportement de l'autre. A tout autre instant du cycle, ils agissent comme deux éléments indépendants. En outre, ces deux composants sont physiologiquement différents de par leur nature et leur fonctionnement. Le ventricule est un muscle qui se contracte et se relâche, atteignant une raideur très élevée au maximum de sa contraction tandis que l'aorte est un vaisseau sanguin qui n'est nullement sujet à une contraction. D'autre part, l'ordre de grandeur des variables du ventricule n'est pas similaire à celui de l'aorte. L'intervalle de variation des valeurs du ventricule est nettement plus étendu que celui de



l'aorte.

Exemple :

la valeur minimale de la pression dans le ventricule est beaucoup plus petite que la valeur minimale de la pression dans l'aorte.

Par conséquent, le raisonnement en termes qualitatifs effectué pour l'analyse du comportement du ventricule est sensiblement différent de celui effectué pour l'aorte.

La duplicité des raisonnements se reflète dans le choix d'un double contexte d'analyse où les valeurs qualitatives prendront des significations différentes : un contexte pour déterminer les valeurs des variables concernant le ventricule et un autre pour les variables concernant l'aorte. Cependant, pour les besoins de l'étude du fonctionnement du coeur, les pressions déterminées dans les différents contextes doivent pouvoir être comparées. En effet, pour déterminer entre autres, la phase dans laquelle se trouve le système, une comparaison entre pression du ventricule et pression de l'aorte doit être effectuée. Ces pressions appartenant à des contextes distincts ont une signification différente et ne peuvent être comparées telles quelles. Il faut les soumettre à une transformation pour les ramener dans un contexte commun, que nous appellerons contexte général, où elles seront effectivement comparables.

#### DETERMINATION DU CONTEXTE GENERAL ET DE L'ECHELLE DE VALEURS QUALITATIVES

Le contexte général a été déterminé à la suite de l'observation du graphe PRESSION-TEMPS (voir figure 6.1) représentant l'évolution de deux fonctions au cours du cycle cardiaque : la pression ventriculaire et la pression aortique. Nous en avons dégagé la valeur minimale selon l'axe des pressions et lui avons donné arbitrairement une valeur qualitative minimale : 0. Nous avons dégagé la valeur maximale du graphe et lui avons attribué une valeur qualitative élevée : G (grand). Nous avons ajouté une valeur qualitative supérieure à celle-ci qui sera la valeur maximale de l'échelle : TG (très grand). Cette valeur a été ajoutée pour le cas où il se produirait des poussées de pression supérieures à la normale. Nous avons découpé l'intervalle ainsi obtenu en intervalles plus réduits dont les bornes ont reçu une valeur qualitative particulière. Le résultat de ce raisonnement



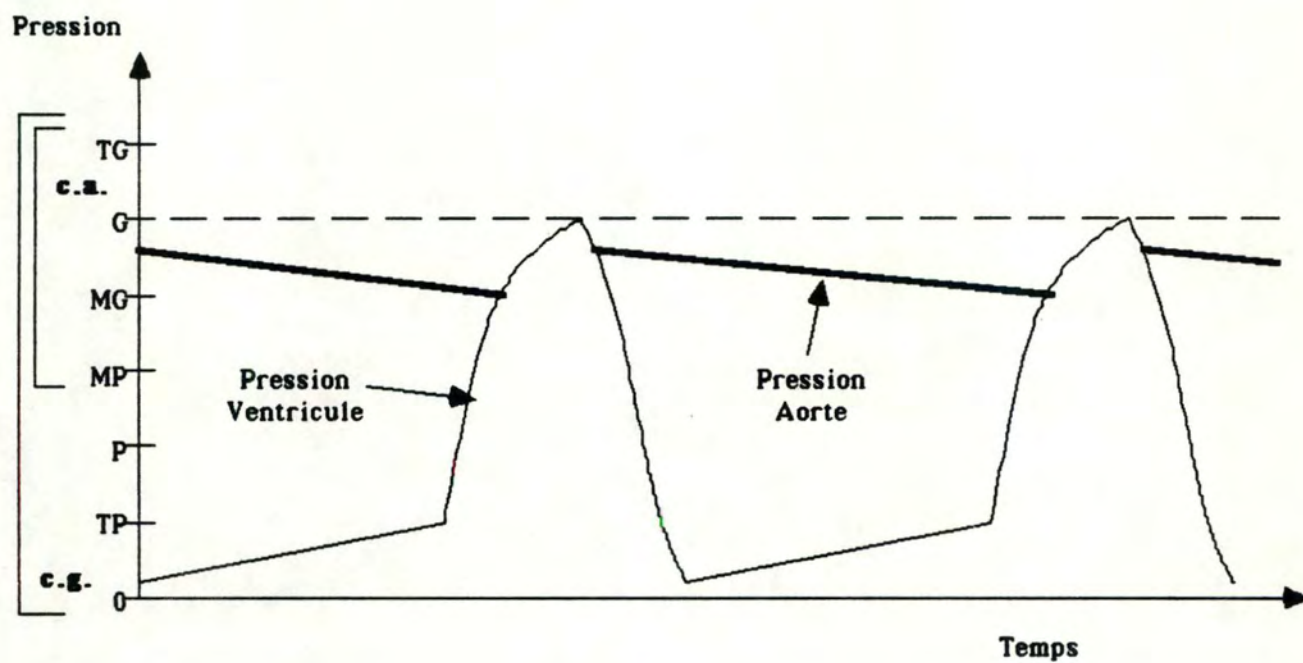


Figure 6.1. : Choix de l'échelle de valeurs qualitatives par observation du graphe PRESSION-TEMPS

nous donne l'échelle de valeurs suivante dont les éléments sont ordonnés de manière strictement croissante :

$$0 < TP < P < MP < MG < G < TG$$

avec TP = très petit

P = petit

MP = moyennement petit

MG = moyennement grand

G = grand

TG = très grand

Cet ensemble ordonné fournit les termes de base pour la détermination des valeurs de tous les paramètres et de toutes les variables du système. Il représente, en quelque sorte, l'ensemble qualitatif des valeurs absolues possibles.

#### DETERMINATION DU CONTEXTE DU VENTRICULE

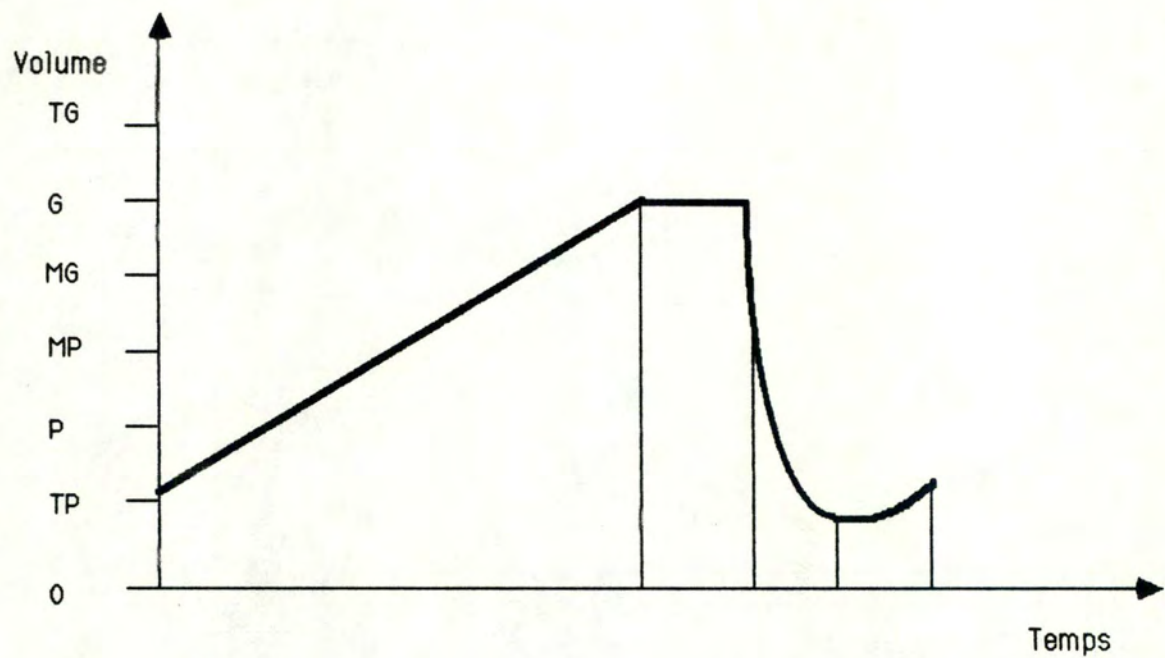
La valeur minimale et maximale du graphe désignent, toutes deux, des valeurs caractérisant la pression dans le ventricule. La signification des valeurs de l'échelle choisie dans le contexte général peut donc être utilisée dans le contexte du ventricule en ce qui concerne les pressions. Pour les volumes, la signification des termes qualitatifs peut être cernée en observant le graphe 6.2. représentant la variation du volume ventriculaire en fonction du temps. Pour les autres variables caractérisant le ventricule, nous ne pouvons que dégager intuitivement leur signification en fonction de leur rôle dans le fonctionnement du système.

contexte général = contexte du ventricule

#### DETERMINATION DU CONTEXTE DE L'AORTE

Dans le graphe PRESSION-TEMPS pour un coeur physiologiquement normal, la courbe correspondant à l'aorte ne descend jamais en dessous d'un certain point de l'axe des ordonnées. Ce point représente plus ou moins la valeur MG selon la signification dans le contexte général de l'échelle qualitative. Ainsi, selon ce contexte, la pression aortique ne varie qu'entre les valeurs MG et G. Nous ne posséderions alors que deux valeurs qualitatives pour définir l'évolution de cette pression, ceci ne





**Figure 6.2 : Signification des valeurs qualitatives en ce qui concerne les volumes de sang dans le ventricule gauche**

peut donner un bon aperçu de sa variation dans le temps.

En considérant la possibilité d'analyser des cas physiologiquement anormaux, l'ensemble des valeurs possibles des pressions de l'aorte doit être étendu au groupe de valeurs : MP - MG - G - TG de l'échelle générale. Nous nous proposons de redécouper cet intervalle de façon à ce qu'il englobe toutes les valeurs qualitatives connues. L'intervalle [MP, TG] dans le contexte général correspondra à l'intervalle [O, TG] dans le contexte de l'aorte.

- contexte général	:	O	TP	P	MP	MG	G	TG			
- contexte de l'aorte :					O	TP	P	MP	MG	G	TG

#### DETERMINATION DES REGLES DE CORRESPONDANCE ENTRE LES 2 CONTEXTES

En raison de la différence de signification des quantités qualitatives des contextes, la comparaison de valeurs issues de contextes différents peut se réaliser moyennant une transformation préalable de ces quantités dans le contexte général. Une correspondance entre signification des valeurs du ventricule et signification des valeurs de l'aorte doit être établie. Notons que la correspondance de signification ne sera déterminée que pour les pressions car selon les règles définies au chapitre 4, les comparaisons entre contextes sont établies uniquement via les pressions. En effet, pour déterminer les phases, il faut comparer la pression du ventricule et celle de l'aorte. Pour calculer la valeur du flux aortique, il faut soustraire la pression aortique de la pression ventriculaire. Aucun autre calcul ne fait intervenir des valeurs de contextes différents.

La signification des valeurs des autres variables du système découlera de leur fonction dans le système et du contexte auquel elles appartiennent.

#### CORRESPONDANCE ENTRE VALEURS DU CONTEXTE DU VENTRICULE ET DE L'AORTE

Cette correspondance a été établie selon la manière dont nous avons déterminé la signification des valeurs des pressions dans le contexte de l'aorte.



- contexte ventricule : O    TP    P    MP    MG    G    TG
- contexte de l'aorte :                    O TP P MP MG G TG

REMARQUE : Il est fort peu probable que la pression aortique descende en dessous de la valeur MP du contexte du ventricule. Néanmoins si cela se produit, nous considérerons la pression comme étant nulle.

#### REGLES DE TRANSFORMATION DES PRESSIONS D'UN CONTEXTE A L'AUTRE

<u>Contexte ventricule</u>	<u>Contexte aorte</u>
O <----->	O
TP <----->	O
P <----->	O
MP <----->	TP
MG <----->	P
	MP
	MG
G <----->	G
TG <----->	TG

Ces règles seront également utiles pour analyser l'impact du flux aortique sur le ventricule d'une part et sur l'aorte d'autre part. Le calcul de ce flux met en rapport les pressions du ventricule et de l'aorte. Dès lors, il faut transformer la pression de l'aorte dans le contexte ventriculaire pour estimer le flot sanguin dans ce contexte. De même, il faut transformer la pression ventriculaire dans le contexte de l'aorte pour estimer le flux dans le contexte de l'aorte.

#### 6. 3. 2 OPERATIONS QUALITATIVES

Les valeurs qualitatives des différentes variables du système sont calculées sur base d'équations tirées du modèle électrique (Chap. 4). Ces équations mettent en oeuvre les opérations arithmétiques courantes : addition, soustraction, multiplication et division. Une redéfinition qualitative de ces opérations, représentée sous forme de tables, est nécessaire pour répondre aux impératifs de notre méthode de raisonnement.

C'est dans le calcul qualitatif que va se refléter le

caractère imprécis de notre raisonnement. En effet, quelle solution donner à une opération dont les arguments ont une signification floue ? Si nous additionnons la valeur qualitative TP à la valeur G, le résultat peut être la valeur TG ou la valeur G. Si nous devons soustraire la valeur P de la valeur P, le résultat pourrait être 0, TP ou encore -TP.

Pour résoudre ce problème, deux solutions sont envisageables. En prenant l'hypothèse qu'une opération ne possède qu'un seul résultat pour un jeu d'arguments, nous pouvons :

Soit, définir plusieurs opérations pour un même opérateur;

Exemple :

pour l'opérateur +,  
nous pouvons avoir 2 opérations :

1. opération :  $O + TP = O$   
                   $TP + TP = TP$   
                   $P + TP = P$   
                  etc....  
2. opération :  $O + TP = TP$   
                   $TP + TP = P$   
                   $P + TP = MP$   
                  etc....

Soit, définir une seule opération pour chaque opérateur.

Exemple :

pour l'opérateur +,  
l'opération unique serait définie comme suit :  
                   $O + TP = TP$   
                   $TP + TP = P$   
                   $P + TP = MP$   
                  etc....

Si plusieurs opérations sont définies pour un même calcul, toutes les solutions possibles qui se présenteraient dans le système réel seraient envisagées. L'analyse d'un tel système sera complète mais assez complexe. En effet, bons nombres de calculs devront être exécutés en chaîne conduisant ainsi à un éclatement combinatoire des solutions assez important.



Exemple :

Calcul du flux passant par la valve mitrale :

$$\text{débitmit} = \frac{\text{Pression réservoir} - \text{pression ventricule}}{\text{Résistance de la valve}}$$

avec Pression ventr. = volume ventr. \* raideur ventr.

Soit 2 opérations pour \*

2 opérations pour -

2 opérations pour %

. le calcul de la pression du ventricule donnera 2 solutions

. le calcul de la différence des pressions donnera 4 solutions si les 2 premières sont envisagées

. le calcul du flux donnera 8 solutions (peut-être pas toutes distinctes) si toutes les solutions des calculs précédents sont envisagées

Si une seule opération est admise pour un calcul, les variables du système ne posséderont qu'une seule solution possible rendant ainsi l'analyse plus aisée mais moins complète. L'opération retenue serait celle qui a la plus grande possibilité de se produire. Cette possibilité est définie intuitivement en fonction du raisonnement de l'expert médical.

La deuxième possibilité faciliterait la compréhension de l'étudiant en réduisant l'ambiguïté qui serait apparue au niveau de l'interprétation des valeurs des variables lors d'une explosion combinatoire.

La solution retenue est une combinaison des deux possibilités présentées ci-dessus. Un même opérateur peut avoir plusieurs opérations. Mais l'opération à appliquer lors d'un calcul qualitatif est déterminée sans ambiguïté. Elle dépend du contexte dans lequel le calcul qualitatif s'effectue et de la signification des variables arguments. Ainsi pour chaque calcul, les opérations à utiliser sont définies et la valeur d'une variable calculée est toujours unique.



Exemple :

- 2 multiplications
  - 1 pour le calcul de la pression ventriculaire
  - 1 pour le calcul de la pression aortique
- 3 divisions
  - 1 pour le calcul du volume ventriculaire
  - 1 pour le calcul du volume aortique
  - 1 pour le calcul des flux

La définition des opérations est présentée sous forme de tables. Un exemple est donné à la figure 6.3 et la figure 6.4 et les autres opérations sont détaillées en annexe 1. Voici la liste de toutes les opérations que nous avons définies.

- Contexte ventricule :
  - calcul pression ventr. : multv-PE
  - calcul volume ventr. : divv-QE
- Contexte aortique :
  - calcul pression aorte : multao-PE
  - calcul volume aorte : divao-QE
- Contexte général :
  - calcul d'une différence de pression : diffn
  - calcul d'un flux : div-PR
  - calcul d'une somme de résistances : addit

La construction de certaines de ces tables d'opérations qualitatives a été effectuée selon un raisonnement physiologique (calcul des pressions et des volumes). En fonction de la signification des variables arguments et de leur valeur, nous avons estimé la valeur de la variable résultat selon son comportement dans le modèle physiologique, faisant ainsi abstraction de toutes les règles algébriques régissant les opérations arithmétiques.

Exemple :

Calcul du volume ventr. = pression ventr. / raideur vent  
Si pression-ventr. = 0 et raideur-ventr. = TP  
Alors volume-ventr. est estimé à TP

Ainsi nos opérations sont toutes des lois de composition interne mais aucune ne forme un groupe abélien, au sens algébrique du terme, sur l'ensemble des valeurs qualitatives.



# **Calcul de la pression du ventricule**      **MULTV-QE**

Volume du ventricule \* Elastance = Pression du ventricule

<div>E Q</div>	0	TP	P	MP	MG	G	TG
0	0	TP	TP	P	P	MP	MP
TP	0	TP	TP	P	MP	MG	MG
P	TP	TP	P	MP	MG	G	TG
MP	TP	P	P	MP	MG	G	TG
MG	TP	P	MP	MP	MG	G	TG
G	P	P	MP	MP	MG	TG	TG
TG	P	MP	MP	MG	G	TG	TG

**Figure 6.3.**

**Calcul de la pression de l'aorte****MULTAO-QE**

Volume de l'aorte \* Elastance = Pression de l'aorte

<div>E Q</div>	0	TP	P	MP	MG	G	TG
0	0	0	TP	TP	TP	P	P
TP	0	TP	TP	P	P	P	MP
P	0	TP	P	P	MP	MP	MG
MP	TP	P	P	MP	MG	MG	G
MG	TP	P	MP	MG	MG	G	G
G	P	MP	MG	G	G	TG	TG
TG	P	MP	MG	G	TG	TG	TG

Figure 6.4.



Définition du groupe abélien :

Si une opération  $\circ$  définie sur un ensemble  $E$

- est une loi de composition interne ( $E \circ E \rightarrow E$ )
- est commutative
- est associative
- possède un neutre
- possède un symétrique

Alors  $E, \circ$  est un groupe abélien

Exemple : (cfr. figure 6.4)

Soit  $V = \{0, TP, P, MP, MG, G, TG\}$

$V$ , multao-QE n'est pas un groupe abélien car

- multao est une loi de composition interne
- multao n'est pas commutative :  $TP * G = P$   
 $G * TP = MP$
- multao n'est pas associative :  $(TP * G) * MG = MP$   
 $TP * (G * MG) = P$
- multao ne possède pas de neutre
- multao ne possède pas de symétrique

La même démonstration peut être faite pour les autres opérations qualitatives définies.

Cette caractéristique de nos opérations nous a contraintes à définir explicitement les opérations inverses dont nous avons besoin : divao-PE, divv-PE, diffn ...)

## 6.4 CALCUL DES VALEURS DES VARIABLES A L'INSTANT SUIVANT

---

### 6.4.1 PRINCIPES

#### PROCESSUS TEMPOREL A 2 VARIABLES

L'évolution du comportement cardiaque est modélisé par un système temporel à deux instants : l'instant courant et l'instant d'après. L'état du système et la valeur de ses variables peuvent être étudiés à un moment donné et/ou au moment suivant, les retours en arrière dans le temps n'étant pas permis.



## PROCESSUS DISCRET

Le fonctionnement du coeur est continu dans le temps. Cependant il n'est pas nécessaire d'observer chaque moment de ce fonctionnement en raison de la faible précision des quantités qualitatives et des besoins de l'utilisateur. En effet, celui-ci ne désire pas étudier en détail l'évolution des variables à chaque instant mais préfère connaître le sens de variation et l'ordre de grandeur de ces variables à certaines étapes caractéristiques du fonctionnement du système.

Le choix des différentes étapes d'analyse a été influencé par notre besoin de donner une appréciation globale de l'évolution du coeur sans s'attarder trop longuement sur des moments sans grand intérêt. Par conséquent, la longueur des intervalles de temps entre deux observations n'est pas constante mais est fonction de l'instant en cours d'observation. La phase de repos, la plus longue dans le battement cardiaque, est sujette à deux étapes d'analyse (au début de cette phase et à la fin) avec un temps très long entre elles-deux, tandis que la contraction fera l'objet de quatre observations avec un temps très court entre celles-ci car les variables évoluent plus rapidement et de manière non monotone contrairement à la phase de repos. [32]

## PROPAGATION DES VALEURS

La propagation dans le temps est orchestrée par la propagation des variables d'état du système et de la variable d'entrée  $N_v$  (nombre d'élastiques autour du coeur à l'instant  $t$ ). Les variables d'état sont les volumes ou pressions du ventricule et de l'aorte. Nous considérons uniquement les volumes pour cette propagation dans le temps sans pour autant nuire à la généralité du système puisque, si on possède les valeurs des pressions, il est possible de calculer les valeurs des volumes. Il y a donc 3 variables à faire évoluer.



#### 6.4.2 CHOIX DES ETAPES ET LONGUEUR DES INTERVALLES

La découpe d'un battement en étapes d'observation a été effectuée selon certains critères :

- les moments d'observations sont dépendants de 3 facteurs :
  1. l'évolution du nombre d'élastiques autour du ventricule
  2. la raideur maximale du ventricule
  3. la durée du rythme cardiaque
- le nombre d'observations sur un battement est toujours le même quelles que soient les valeurs des paramètres et variables

Etant donné ces hypothèses, nous avons procédé comme suit :

1. Nous avons découpé le temps de battement en 3 intervalles primaires correspondant chacun à un état d'évolution du nombre d'élastiques autour du ventricule
  - intervalle 1 : nombre d'élastiques minimal ( $N_v = 0$ ) : repos
  - intervalle 2 : nombre d'élastiques augmentant : contraction
  - intervalle 3 : nombre d'élastiques diminuant : relaxation

Ces intervalles sont de différentes durées. Selon les analyses physiologiques :

$$\begin{aligned} \text{intervalle 1} &\neq \text{intervalle 2} + \text{intervalle 3} \\ \text{intervalle 2} &\neq \text{intervalle 3} \end{aligned}$$

La longueur de ces intervalles varie selon le rythme cardiaque :

$$\begin{aligned} \text{intervalle 1 pour un rythme petit} &< \text{intervalle 1 pour un} \\ \text{rythme normal} &< \text{intervalle 1 pour un rythme long} \\ i &= 1, 2 \text{ ou } 3 \end{aligned}$$

2. Nous avons découpé chaque intervalle primaire en intervalles secondaires dont chaque borne est une valeur qualitative dépendante de la raideur maximale du ventricule et du nombre d'élastiques autour du ventricule.

Exemple :

l'intervalle 2 comprend 3 intervalles secondaires

Pour une raideur maximale normale

les intervalles secondaires sont :  $[P, MP[$  ;  $[MP, MG[$  ;  $[MG, G]$

Les durées de ces intervalles secondaires, une fois encore, ne sont pas identiques. Elles dépendent de leur intervalle primaire et du rythme cardiaque. La somme des durées des inter-

valles secondaires est égale approximativement à l'intervalle primaire correspondant.

Les schémas de la figure 6.5 (a et b) donnent toutes les valeurs possibles des durées des intervalles de temps.

#### 6.4.3. PROPAGATION DES VARIABLES D'ETAT

##### VOLUMES

Selon la règle d'EULER d'intégration appliqué ici aux volumes :

$$\text{VOL}(t+\Delta t) = \text{VOL}(t) + \frac{d\text{VOL}(t)}{dt} * \Delta t$$

Transformons cette règle :

$$- \text{VOL VENT.}(t+\Delta t) = \text{VOL VENT.}(t) + \text{Variation VOL VENT.}(t) * \Delta t$$

où  $\Delta t$  = longueur de l'intervalle entre le moment  $t$  et le moment  $t + \Delta t$

(cette longueur est déterminée selon le tableau présenté ci-avant)

$$\text{Variation VOL VENT.}(t) = \text{débit valve mitrale} - \text{débit valve aort.}$$

$$- \text{VOL AORTE}(t+\Delta t) = \text{VOL AORTE}(t) + \text{Variation VOL AORTE}(t) * \Delta t$$

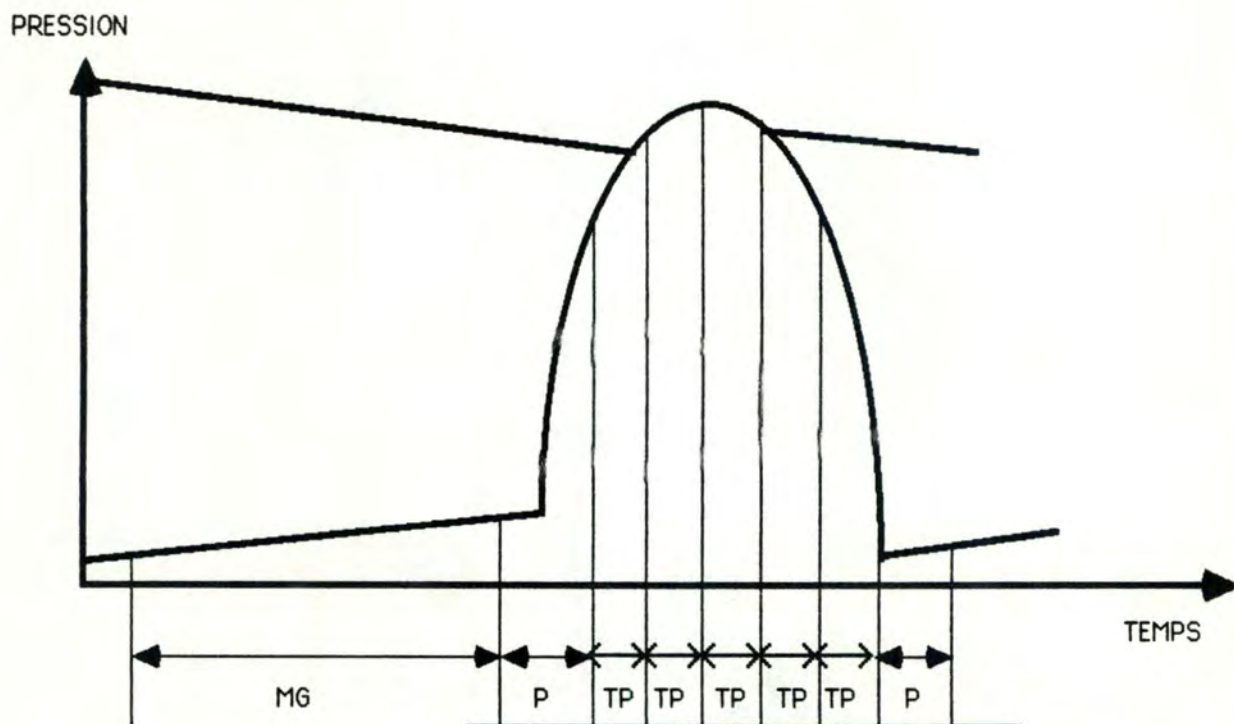
où  $\Delta t$  =  $\Delta t$  de la règle précédente

$$\text{Variation VOL AORTE}(t) = \text{débit valve aort.} - \text{debit réseau périp.}$$

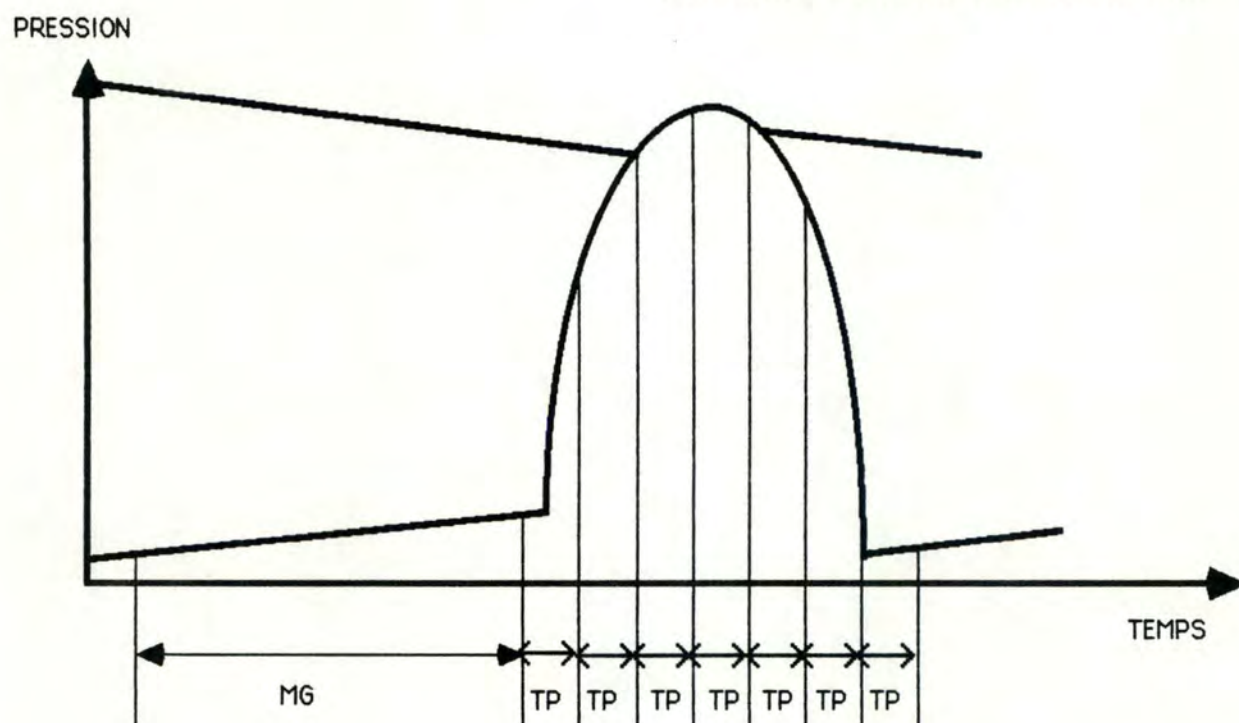
##### RAIDEUR DU VENTRICULE

Le nombre d'élastiques autour du ventricule à l'instant  $t+\Delta t$  dépend du nombre d'élastiques à l'instant  $t$ , de son sens de variation et de la raideur maximale du ventricule. Cette valeur est déterminée conformément au tableau présenté ci-dessus.



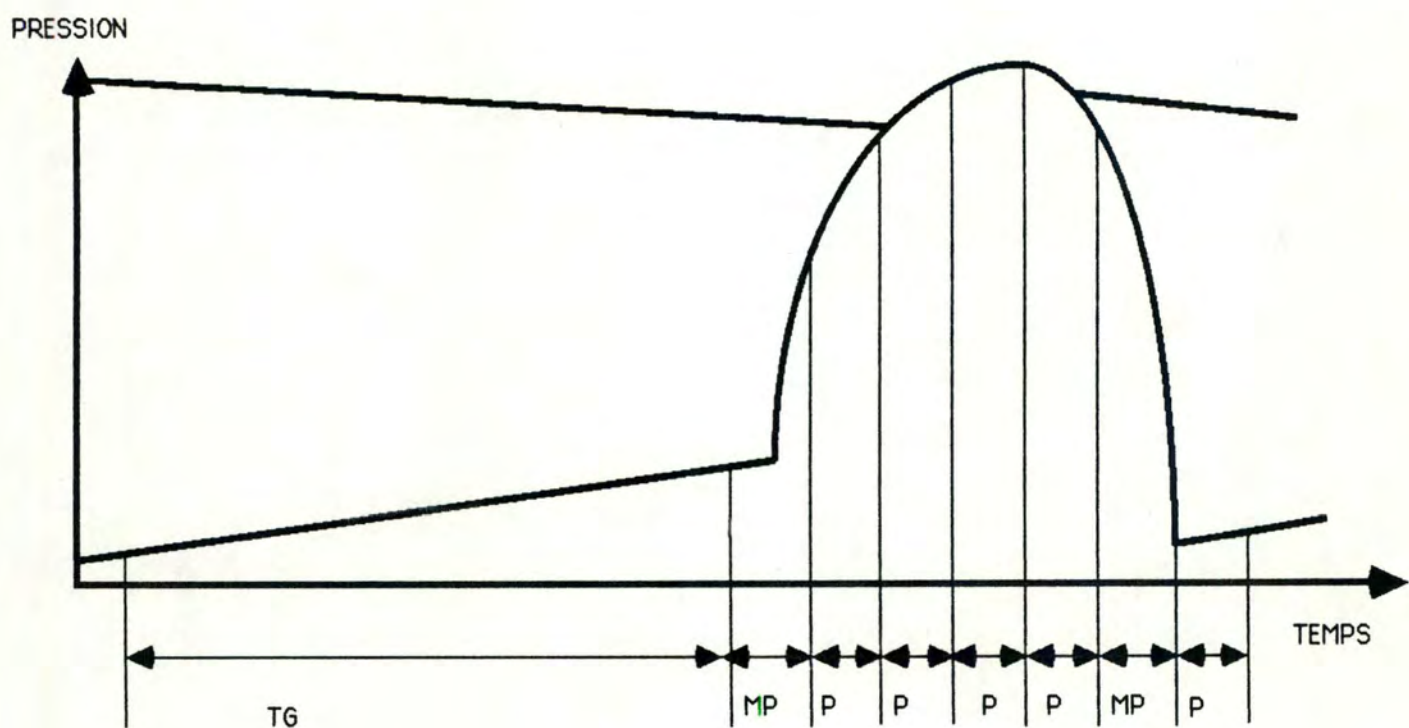


### RYTHME CARDIAQUE NORMAL



### RYTHME CARDIAQUE COURT

FIGURE 6.5 a : Choix de la longueur des intervalles de temps entre 2 observations suivant que le rythme cardiaque est normal ou court



### RYTHME CARDIAQUE LONG

FIGURE 6.5 b : Choix de la longueur des intervalles de temps entre 2 observations si le rythme cardiaque est long



Example :

$$S_1 \quad N_V(t) = P$$

sens de variation des  $N_v$  = croissant

Emax est normal

Alors  $Nv(t+\Delta t) = MP$

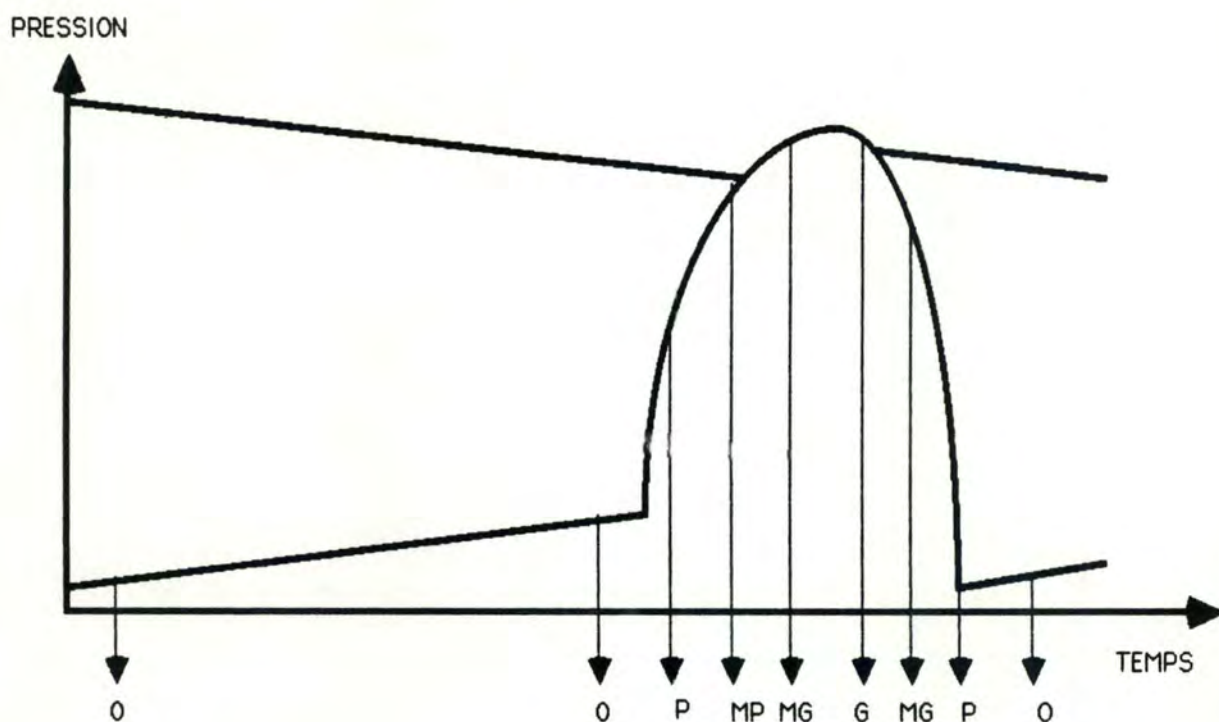
Les schémas de la figure 6.6 (a et b) donnent toutes les valeurs possibles pour les Nv.

#### 6.4.4 OPERATIONS QUALITATIVES

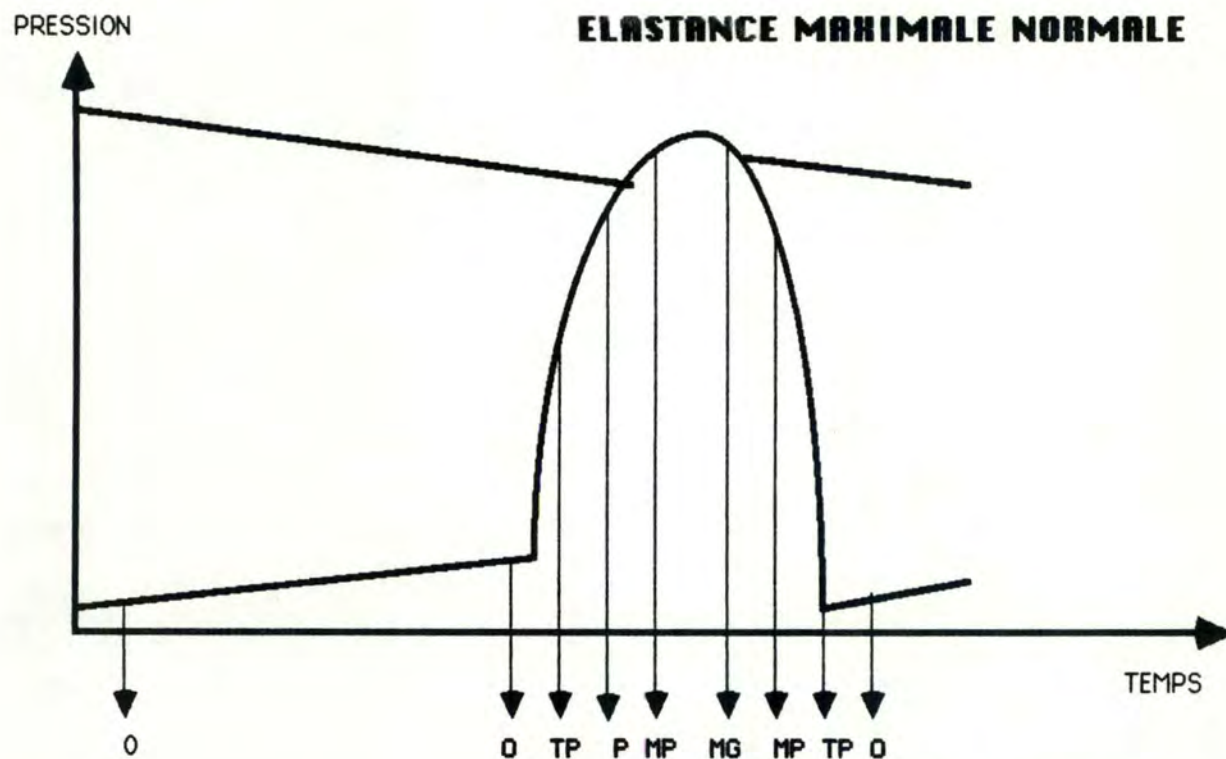
Ici aussi, la différence des contextes est primordiale pour déterminer les opérations utiles à la propagation des valeurs des volumes et pour les mêmes raisons qu'explicitées au paragraphe 6.3.

Suivant la formule d'EULER, 4 opérations sont définies (les tables sont reprises en annexe 1) :

- ```
- pour le ventricule : multv
                        additv
- pour l'aorte       : multao
                        additao
```



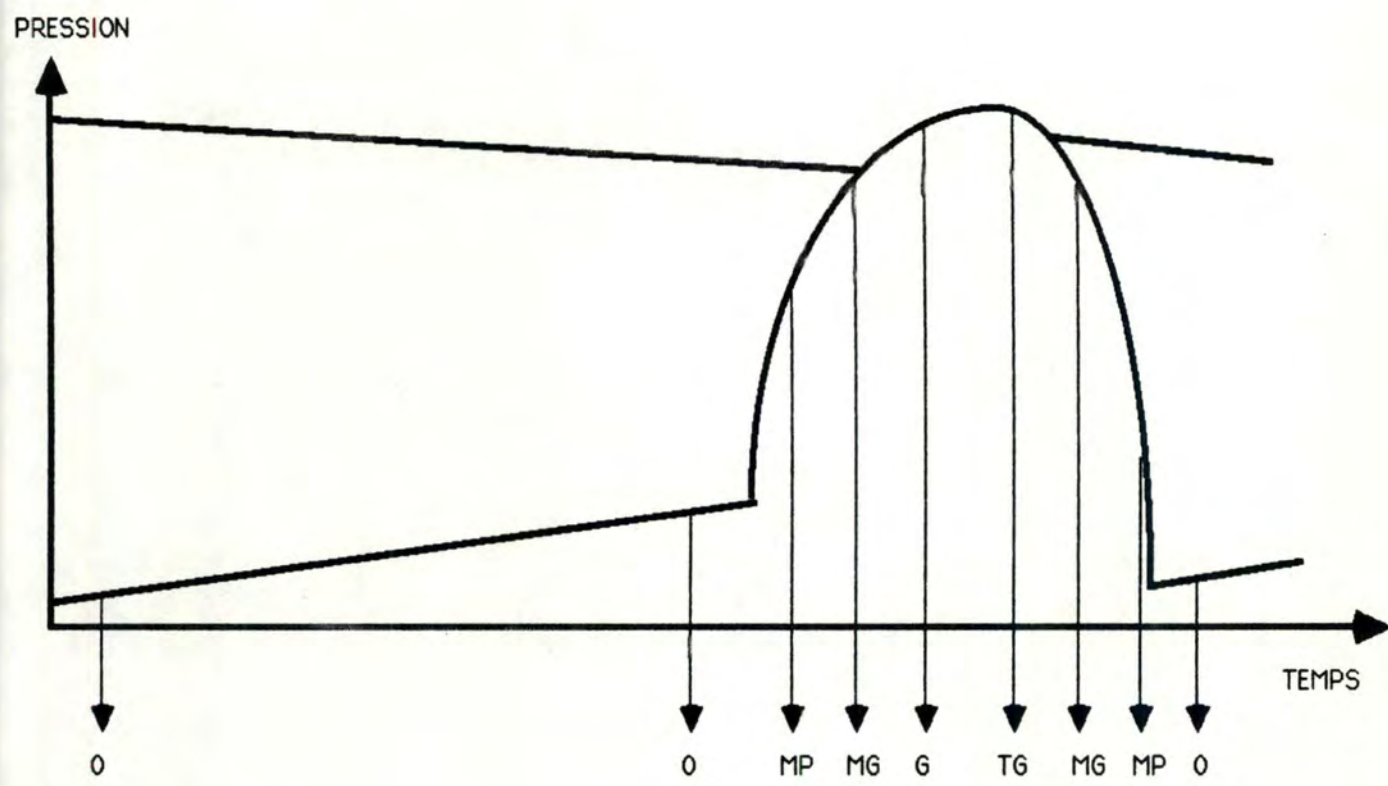
### ELASTANCE MAXIMALE NORMALE



### ELASTANCE MAXIMALE PETITE

Figure 6.6.a. : Choix de la valeur du nombre d'élastiques autour du coeur à chaque observation suivant que l'élastance maximale du ventricule est normale ou petite





### ELASTANCE MAXIMALE GRANDE LONGUE

Figure 6.6.b. : Choix de la valeur du nombre d'élastiques autour du coeur si l'élastance maximale du ventricule est grande

## CHAPITRE 7

### IMPLEMENTATION MICRO-PROLOG

#### 7.1 INTRODUCTION

---

Les éléments qui nous ont incitées à utiliser micro-Prolog comme langage d'implémentation de notre système sont essentiellement de nature pratique.

A l'origine nous avions l'intention de formaliser notre problème à l'aide d'un outil de programmation logique. Nous savions, d'autre part, que le système que nous développerions devrait fonctionner sur IBM-PC ou compatibles. Sous cette contrainte, la seule possibilité qui initialement s'offrait à nous pour utiliser la programmation logique dans notre démarche de conception, consistait à utiliser le langage micro-Prolog.

Par ce présent chapitre, notre intention n'est pas d'exposer en détail l'étape d'implémentation du système en micro-Prolog (programme en Annexe 3). Au contraire, nous désirons uniquement attirer l'attention sur les principales caractéristiques de la phase de programmation. Nous terminons par une rapide critique de l'outil et de la programmation logique en général.

#### 7.2 PRESENTATION DES PRINCIPALES CARACTERISTIQUES DU PROGRAMME MICRO-PROLOG

---

##### 7.2.1 PRINCIPE GENERAL D'UNE EXECUTION

L'idée de cette section est de présenter brièvement au lecteur les différentes phases possibles d'une exécution. Le



détail de chacune d'elle est exposé dans les sous-sections qui suivent.

La première phase consiste à initialiser le système afin que l'exécution se déroule correctement. L'initialisation exige une participation de l'utilisateur. Celui-ci doit, en effet, intégrer au système les valeurs de quelques paramètres et variables qu'il connaît et qu'il souhaite utiliser comme base de son étude.

Une fois le système initialisé, l'utilisateur dispose de trois possibilités d'exploitation. Il peut, tout d'abord, stopper l'exécution et dans ce cas ne rien apprendre de plus du système. Cependant, une approche plus intéressante pour l'utilisateur est de conduire, au choix, l'exécution vers une deuxième phase qui lui permet de connaître l'état courant du système ou encore vers une troisième phase qui lui permet de faire évoluer le système dans le temps.

Ainsi, après l'initialisation, si l'utilisateur désire étudier l'évolution du système (c'est-à-dire du coeur gauche sur un ou plusieurs cycles cardiaques), il lui suffit d'abord d'observer le système dans son état courant, de le propager ensuite à l'instant suivant et de recommencer ces deux opérations autant de fois qu'il le souhaite.

Nous retiendrons donc trois phases principales dans l'exécution : l'initialisation du système, la détermination de l'état du système à l'instant courant et la propagation dans le temps du système de l'instant courant à l'instant suivant. Ces phases sont détaillées dans les trois sous-sections présentées ci-dessous.

## 1. INITIALISATION DU SYSTEME

Afin de généraliser la procédure d'initialisation et de ne pas compliquer outre mesure la tâche de l'utilisateur dans cette phase, nous considérerons toujours qu'une exécution commence lorsque le coeur gauche est en début de phase de repos. L'utilisateur doit transmettre interactivement au système les valeurs des paramètres sur lesquelles il désire travailler ainsi que les valeurs des pressions et/ou des volumes ventriculaires et aortiques qu'il connaît.



Les paramètres pour lesquels l'utilisateur doit donner une valeur à l'initialisation sont :

- \* la pression dans le réservoir
- \* la résistance du réseau systémique
- \* la résistance passante de la valve mitrale
- \* la résistance bloquante de la valve mitrale
- \* la résistance passante de la valve aortique
- \* la résistance bloquante de la valve aortique
- \* l'élastance maximale du ventricule gauche
- \* la raideur de l'aorte
- \* le rythme cardiaque

Les variables du système auxquelles l'utilisateur devrait donner une valeur à l'initialisation sont :

- \* la pression dans le ventricule
- \* le volume de sang dans le ventricule
- \* la pression dans l'aorte
- \* le volume de sang dans l'aorte

Il est à noter que lors de l'initialisation l'utilisateur peut recourir aux indications proposées par le système en matière de valeurs possibles des paramètres et des variables.

Après avoir initialisé le système, l'utilisateur peut, comme il l'a été signalé plus haut, outre stopper l'exécution, connaître l'état du système à l'instant courant ou propager le système dans le temps de l'instant courant à l'instant suivant.

Remarque : Seule la phase d'initialisation permet à l'utilisateur de fournir des valeurs aux paramètres et aux variables du système énumérés ci-dessus. Pour cette raison toute



modification des valeurs de ces paramètres ou variables devra être réalisée lors de l'initialisation. Ceci signifie qu'il est impossible, en cours d'exécution, de modifier une valeur transmise au système à l'initialisation et que si l'utilisateur désire travailler sur d'autres valeurs, il doit stopper l'exécution en cours et réinitialiser le système avec les nouvelles valeurs pour lesquelles il souhaite observer le comportement du système. Cette contrainte peut paraître assez lourde à première vue, mais en y réfléchissant de plus près, il semble préférable pour l'utilisateur de pouvoir observer l'impact de modifications sur le système dès le début d'un cycle cardiaque.

## 2. DETERMINATION DE L'ETAT DU SYSTEME A L'ETAT COURANT

L'utilisateur peut interroger la base de connaissances du système s'il souhaite s'informer de l'état courant du coeur gauche. Il obtiendra une réponse aux questions suivantes :

- \* Quel est le volume de sang dans le ventricule et quel est le sens d'évolution de ce volume?
- \* Quel est le volume de sang dans l'aorte et quel est le sens d'évolution de ce volume?
- \* Quelle est la pression dans le ventricule gauche et quel est le sens d'évolution de cette pression?
- \* Quelle est la pression dans l'aorte et quel est le sens d'évolution de cette pression?
- \* La valve mitrale est-elle ouverte?
- \* La valve mitrale est-elle fermée?
- \* La valve aortique est-elle ouverte?
- \* La valve aortique est-elle fermée?
- \* Quel est le débit de la valve mitrale?
- \* Quel est le débit de la valve aortique?

- \* Quel est le débit vers le réseau systémique?
- \* Quelle est la variation du volume de sang dans le ventricule?
- \* Quelle est la variation du volume de sang dans l'aorte?
- \* Dans quelle phase du cycle cardiaque se trouve le coeur gauche?
- \* Quelles sont les valeurs des paramètres introduites à l'initialisation?
- \* Le ventricule gauche se remplit-il?
- \* Le ventricule gauche éjecte-t-il?

### 3. PROPAGATION DU SYSTEME DE L'INSTANT COURANT A L'INSTANT SUIVANT

L'utilisateur, s'il veut propager le système dans le temps, doit le demander explicitement en interrogeant la base de connaissances. Le système est propagé de l'instant courant à l'instant suivant via les volumes sanguins et le "nombre d'élastiques autour du coeur". L'instant courant est modifié et l'utilisateur peut à nouveau interroger la base de connaissances, comme au point 2 de la section 7.2.1 afin d'observer les nouvelles caractéristiques du système propagé.

### 7.2.2 CONTENU DE LA BASE DE CONNAISSANCES

#### 1. LES FAITS

##### FAITS RENSEIGNANT DE LA VALEUR DES PARAMETRES DU SYSTEME

Le fait "Rmitrale-pas(0)" signifie que la résistance passante de la valve mitrale est nulle.

Le fait "Rper(MG)" signifie que la résistance du réseau périphérique est "moyennement grande".



## FAITS RENSEIGNANT DE LA VALEUR DE CERTAINES VARIABLES DU SYSTEME

Le fait "Nv(0)" signifie que le "nombre d'élastiques autour du coeur" est minimum.

En particulier pour cette variable, un fait indique dans la base son sens d'évolution. Ainsi, "AugmNv(+)" signifie que le "nombre d'élastiques autour du coeur" augmente.

## FAITS REPRESENTANT LES OPERATIONS QUALITATIVES DEFINIES PAR LES TABLES

Chaque fait donne le résultat d'une opération qualitative particulière pour deux valeurs d'arguments données et prises dans l'ensemble des valeurs qualitatives {0, TP, P, MP, MG, G, TG}.

L'opération "multao-QE" qui permet de déterminer la pression dans l'aorte par le produit du volume de sang dans l'aorte et de la raideur de celle-ci, est définie par la figure 7.1 et par l'ensemble des faits de la figure 7.2.

## FAITS PERMETTANT DE SITUER L'ETAT DU COEUR GAUCHE DANS LE CYCLE CARDIAQUE

Le fait "repos(dedut)" signifie que le coeur gauche est en début de phase de repos.

Le fait "ejec(oui)" signifie que le ventricule gauche éjecte du sang dans l'aorte.

| $I_a \Delta t$ | TP  | P   | MP  | MG  | G   | TG  |
|----------------|-----|-----|-----|-----|-----|-----|
| -TG            | -TG | -TG | -TG | -TG | -TG | -TG |
| -G             | -G  | -G  | -G  | -TG | -TG | -TG |
| -MG            | -MG | -MG | -MG | -G  | -G  | -G  |
| -MP            | -MP | -MP | -MP | -MG | -MG | -G  |
| -P             | -P  | -P  | -P  | -MP | -MP | -MP |
| -TP            | -TP | -P  | -P  | -P  | -P  | -MP |
| 0              | 0   | 0   | 0   | 0   | 0   | 0   |
| TP             | TP  | P   | P   | P   | P   | MP  |
| P              | P   | P   | P   | MP  | MP  | MP  |
| MP             | MP  | MP  | MP  | MG  | MG  | G   |
| MG             | MG  | MG  | MG  | G   | G   | G   |
| G              | G   | G   | G   | TG  | TG  | TG  |
| TG             | TG  | TG  | TG  | TG  | TG  | TG  |

Figure 7.1.

((multao -TG TP -TG))  
 ((multao -TG P -TG))  
 ((multao -TG MP -TG))  
 ((multao -TG MG -TG))  
 ((multao -TG G -TG))

((multao -TG -TG -TG))  
 ((multao -G TP -G))  
 ((multao -G P -G))  
 ((multao -G MP -G))  
 ((multao -G MG -TG))

etc...

Figure 7.2.



## 2. LES REGLES

### REGLES DEGAGEES APRES DIALOGUE AVEC L'EXPERT

Une série de règles permettent de déterminer les phases du cycle cardiaque, la position des valves du coeur gauche, le sens d'évolution des volumes et des pressions, et le débit de la valve mitrale.

La règle de la figure 7.3 signifie que le coeur est en phase de contraction isovolumétrique si la valve aortique est fermée et si le coeur gauche est en activité croissante.

phase(contrac-iso)if aortique-fermee and  
activite-croiss

7.3

### REGLES DEGAGEES SUITE AU DIALOGUE AVEC L'EXPERT ET SUITE AUX RESULTATS D'UNE REFLEXION SUR LA REALISATION D'UN MODELE QUALITATIF DU FONCTIONNEMENT DU COEUR GAUCHE SUR PLUSIEURS CYCLES

Certaines règles n'intègrent pas le mécanisme de propagation du temps et permettent de déterminer le volume de sang dans l'aorte et dans le ventricule en fonction du contexte, la pression dans l'aorte et dans le ventricule en fonction du contexte, le débit de la valve aortique en fonction du contexte. Elles réalisent encore la transformation des valeurs de pression d'un contexte à l'autre. Ainsi, le débit de la valve aortique dans le contexte de l'aorte peut être calculé par la règle (1) de la figure 7.4. La règle (2) de cette même figure convertit une pression aortique du contexte général dans le contexte de l'aorte.

```

debit-aortique-cont-aorte(X) if Pv-cont-aorte(Y) and
                                Paconnu(Z) and
                                plus-petit (Y Z) and
                                Raorte-bloc(x) and
                                diffn(X Y y) and
                                div-PR(y x X) and
                                /

```

(1)

```

Pa-cont-vent(MG) if Pa(MP) and
                                /

```

(2)

7.4

D'autres règles intègrent le mécanisme de propagation du temps et permettent de déterminer en fonction du contexte les volumes de sang dans l'aorte et le ventricule après propagation du temps à l'instant suivant; de déterminer le "nombre d'élastiques autour du coeur après propagation du temps à l'instant suivant ainsi que l'intervalle de temps à considérer lors de la propagation du temps. Le volume de sang dans l'aorte après propagation peut être obtenu par la règle de la figure 7.5.

```

Volao-apres(X) if Volao(x) and
                   Iaorte(Z) and
                   dt(x) and
                   multao(Z x y) and
                   additao (Y y X)

```

7.5

REGLES PERMETTANT A L'UTILISATEUR D'INTERROGER LA BASE DE  
CANNAISSANCES PLUS FACILEMENT

La règle de la figure 7.6 permet à l'utilisateur de demander la pression dans le ventricule et son sens d'évolution par la simple question "is(pres-vent)". Les règles habituelles de détermination de la pression ventriculaire et de son sens d'évolution nécessitent les deux questions plus complexes :  
"which(x:Pv(x))" et "which(x:evolutionPv(x))".



pres-vent if Pv(x) and  
 evolutionPv(y) and  
 PP(x) and  
 PP(sens variation : y) and  
 /  
7.6

## REGLES PERMETTANT LA COMPARAISON DE VALEURS QUALITATIVES

La règle de la figure 7.7 permet de comparer des valeurs qualitatives et en particulier signifie que X est plus petit que Y lorsque le résultat de la différence Y-X est positif.

plus-petit(X Y) if diffn(Y X Z) and  
 pos(Z)  
7.7

### 7.2.3 MODIFICATION DE LA BASE DE CONNAISSANCES

Afin de présenter une chronologie des modifications subies par la base de connaissances, nous les envisagerons dans le cadre du principe général d'une exécution.

#### MODIFICATION DE LA BASE DE CONNAISSANCES A L'INITIALISATION

Comme il l'a été signalé auparavant, toute exécution commence lorsque le coeur gauche est en début de phase de repos. Dans cette situation le "nombre d'élastiques autour du coeur" est minimum et constant, le ventricule gauche n'éjecte pas. Automatiquement, sans l'intervention de l'utilisateur, des faits mentionnant ces caractéristiques sont introduits dans la base de connaissances. Ainsi, les deux faits "Nv(0)" et "AugmNv(0)" indiquent que le "nombre d'élastiques autour du coeur est minimum et constant, le fait "repos(debut)" indique que le coeur est en début de phase de repos et le fait "ejec(non)" indique que le ventricule gauche n'éjecte pas de sang dans l'aorte.

De plus, en cours d'initialisation, l'utilisateur complète la base de connaissances des faits déterminant la valeur des paramètres ainsi que des pressions et des volumes qu'il connaît.



## MODIFICATIONS DE LA BASE DE CONNAISSANCES LORS DE LA DETERMINATION DE L'ETAT DU SYSTEME A L'INSTANT COURANT

Lorsqu'il désire obtenir des indications sur l'état du système à l'instant courant, l'utilisateur ne fait qu'interroger la base de connaissances. Cette phase de détermination de l'état courant du système nécessite simplement un traitement sans modification des informations existant dans la base. Et celle-ci demeure donc inchangée.

## MODIFICATIONS DE LA BASE DE CONNAISSANCES LORS DE LA PROPAGATION DU SYSTEME DANS LE TEMPS

Lorsque l'utilisateur demande que le système soit propagé dans le temps, seules les variables d'état, à savoir les volumes sanguins, et la variable d'entrée qui correspond au "nombre d'élastiques autour du coeur" évoluent effectivement de l'instant courant à l'instant suivant. Ainsi, la propagation du système dans le temps implique uniquement les modifications suivantes de la base de connaissances :

- \* suppression de l'ancienne valeur du volume de sang dans le ventricule et insertion de la nouvelle valeur calculée de ce volume
- \* suppression de l'ancienne valeur du volume de sang dans l'aorte et insertion de la nouvelle valeur calculée de ce volume
- \* suppression de l'ancienne valeur du "nombre d'élastiques autour du coeur" et insertion de la nouvelle valeur calculée de ce nombre
- \* suppression du sens de variation du "nombre d'élastiques autour du coeur" si celui-ci est modifié par la propagation et insertion d'une nouvelle valeur du sens de variation de ce nombre



#### 7.2.4 LES ARTIFICES DE PROGRAMMATION NECESSAIRES A L'OBTENTION DE REPONSES CORRECTES DE LA PART DU SYSTEME

1. Lorsque le coeur gauche est en phase d'éjection, la pression dans l'aorte ne doit plus être déterminée par la règle de production habituelle. Cette pression doit simplement être égale à celle régnant dans le ventricule et ceci durant toute la phase d'éjection. Pour cette raison, dès que le coeur gauche entre en éjection c'est-à-dire dès que la valve aortique est ouverte, un fait indiquant que le ventricule gauche éjecte, à savoir "ejec(oui)", est ajouté à la base de connaissances alors qu'un autre fait indiquant que le ventricule gauche n'éjecte pas, à savoir "ejec(non)" est supprimé de la base.

Dès la fin de l'éjection, lorsque le coeur passe en phase de relaxation, le fait "ejec(oui)" est supprimé à son tour et le fait "ejec(non)" est, quant à lui, réinséré dans la base.

2. Lors de la propagation dans le temps des variables d'état du système, il est nécessaire de déterminer la phase dans laquelle se trouve le coeur gauche afin de calculer à tout instant des valeurs correctes pour les pressions dans le ventricule et dans l'aorte. En effet, comme il l'a été mentionné ci-dessus, en phase d'éjection la pression dans l'aorte doit être égale à la pression dans le ventricule gauche. Pour qu'il en soit ainsi, le fait "ejec(oui)" est inséré dans la base de connaissances dès que le coeur gauche entre en phase d'éjection.

Lorsque la propagation du temps fournit des valeurs de variables d'état qui indiquent que le coeur gauche passe en phase d'éjection, le système ne l'a pas encore détecté. Le fait "ejec(oui)" n'a donc pas été introduit dans la base. Dans ce cas, la pression dans l'aorte est toujours déterminée par la règle de production habituelle puisqu'aucun fait ne signale que le ventricule gauche éjecte.

Afin d'éviter ce problème du calcul de pression en phase d'éjection, la simple propagation des variables d'état est complétée par la détermination de la phase du système. Celle-



ci a pour effet de vérifier que la valve aortique est ouverte lorsque le coeur gauche entre en phase d'éjection et d'insérer le fait "ejec(oui)" dans la base si c'est le cas. Ainsi, la pression dans l'aorte, lors de l'éjection, est toujours déterminée en fonction de la pression dans le ventricule.

3. Lors de la phase d'éjection ventriculaire et plus exactement en fin de cette éjection, la résistance passante de la valve aortique est augmentée afin que le débit de cette valve corresponde mieux au réel perçu. Cette résistance doit, cependant, être rétablie à sa valeur antérieure dès que le coeur gauche passe en phase de relaxation.

Ces augmentations et diminutions de la résistance passante de la valve aortique nécessitent un artifice de programmation dans certains cas.

En effet, si en fin d'éjection la valeur de cette résistance est nulle ou égale à "très petit", l'augmentation a pour effet d'amener cette valeur à "petit". Dans ce cas, pour réaliser une diminution correcte de la valeur de cette résistance en début de relaxation, il faut avoir conservé une trace de sa valeur initiale.

Le problème se pose de la même manière lorsqu'en fin d'éjection la résistance de la valve a la valeur "grand" ou "très grand" dans la mesure où l'augmentation conduit cette valeur à "très grand", valeur maximale possible.

C'est pourquoi, lors de l'augmentation d'une résistance passante de la valve aortique dont la valeur initiale est nulle ou grande, un fait renseignant de cette valeur est introduit dans la base de connaissances. Ce fait permet par la suite une diminution correcte de la valeur de la résistance et est supprimé de la base après que la diminution ait été réalisée.

4. Après discussion avec l'expert, il est apparu que la pression dans le ventricule gauche pouvait être calculée à partir du volume de sang existant dans ce ventricule et inversement. Il en va de même en ce qui concerne le calcul de ces variables dans le référentiel de l'aorte.



Puisque les pressions et volumes ventriculaires et aortiques se déterminent d'une manière identique, nous considérons le contexte du ventricule gauche pour exposer le problème qui nous a conduit à transformer les formules de calcul de ces variables.

La pression dans le ventricule est établie par le produit du volume de sang dans le ventricule et de l'élastance du ventricule.

Le volume de sang dans le ventricule, quant à lui, est déterminé comme le quotient de la division de la pression par l'élastance.

Ces deux définitions ont donné lieu aux deux règles de production de la figure 7.8.

|                      |                      |
|----------------------|----------------------|
| Pv(x) if Volv(y) and | Volv(x) if Pv(y) and |
| Nv(z) and            | Nv(z) and            |
| multv-QE(y z x)      | divv-PE(y z x)       |
| (1)                  | (2)                  |

#### 7.8

De telles définitions engendrent un cycle lorsque la pression ou le volume de sang dans le ventricule doit être déterminé. Il est donc impossible de conserver telles quelles ces règles de production.

L'idée retenue afin d'éliminer ces définitions cycliques s'appuie sur la constatation qu'il existe deux types de pression et de volume. Les valeurs de ces variables sont soit connues soit inconnues. Dans certains cas, en effet, l'utilisateur connaît la pression et/ou le volume et peut transmettre au système une valeur pour ces variables. Nous dirons que ces valeurs sont connues du système. Dans d'autres cas, cependant, l'utilisateur ne connaît pas la pression et/ou le volume et il ne peut transmettre de valeurs de ces variables au système. Nous dirons que ces valeurs sont inconnues du système.

Une fois conscientes des deux types de pression et volume existant, nous pouvons reconsidérer les deux règles de production initiales. En fait, si la pression dans le ventricule est inconnue, il est possible de la déterminer par l'intermédiaire du volume de sang connu dans le ventricule. Inverse



ment, si le volume de sang dans le ventricule est inconnu, il est possible de le déterminer par l'intermédiaire de la pression connue dans le ventricule. Ainsi les règles de production initiales peuvent être remplacées par celles de la figure 7.9.

|                      |                           |
|----------------------|---------------------------|
| Pv(x) if Volv(y) and | Volv(x) if Pvconnu(y) and |
| Nv(z) and            | Nv(z) and                 |
| multv-QE(y z x)      | divv-PE(y z x)            |
| (1)                  | (2)                       |

### 7.9

Envisageons maintenant, pratiquement, les combinaisons possibles de pression/volume connus et/ou inconnus ainsi que leur impact sur la base de connaissances et sur la détermination de ces variables.

- A) L'utilisateur connaît la pression et le volume de sang dans le ventricule. Dans ce cas, les deux faits de la figure 7.10 ont été introduits dans la base de connaissances à l'initialisation de sorte que la détermination de la pression ou du volume ne nécessite pas l'utilisation des règles de la figure 7.9.

|            |         |
|------------|---------|
| Pvconnu(P) | Volv(G) |
| (1)        | (2)     |

### 7.10

- B) L'utilisateur connaît la pression mais ne connaît pas le volume de sang dans le ventricule. Dans ce cas, le fait (1) de la figure 7.10 a été introduit dans la base de connaissances à l'initialisation. La pression dans le ventricule est directement donnée par ce fait tandis que le volume de sang dans le ventricule est déterminé par la règle (2) de la figure 7.9.
- C) L'utilisateur ne connaît pas la pression mais connaît le volume de sang dans le ventricule. Dans ce cas, le fait (2) de la figure 7.10 a été introduit dans la base de connaissances à l'initialisation. Le volume de sang dans le ventricule est donné directement par ce fait tandis que la pression est déterminée par la règle (1) de la



figure 7.9.

- D) L'utilisateur ne connaît ni la pression ni le volume de sang dans le ventricule. Dans ce cas, aucun fait n'a été introduit dans la base de connaissances à l'initialisation et les deux règles de la figure 7.9 qui seront activées échoueront puisque ni  $Pvconnu(y)$  ni  $Volv(y)$  n'existent dans la base.

#### 7.2.5 LES PROBLEMES INHERANTS A LA PROGRAMMATION LOGIQUE

1. L'ordonnancement des règles peut dans certains cas fournir des résultats éronnés et doit donc être assuré avec soin.

##### Exemple

Le débit de la valve aortique dans le contexte du ventricule peut être déterminé par deux règles différentes comme l'indique la figure 7.11. L'une vérifie que le ventricule gauche n'éjecte pas plus de sang dans l'aorte qu'il n'en contient; l'autre suppose que le volume restant dans le ventricule est suffisant pour que l'éjection se déroule correctement. Il est dans ce cas nécessaire de placer la règle la plus restrictive en première position. Si celle-ci échoue, la deuxième pourra être satisfaite sans risque d'incohérence dans le résultat.

```

debit-aortique-cont-vent(x) if phase(contrac-ejec) and
                                Pvconnu(y) and
                                Psys(z) and
                                diffn(y z X) and
                                Raorte-pas(Y) and
                                Rcar(Z) and
                                addit(Y Z x1) and
                                div-PR(X x1 y1) and
                                Volv(z1) and
                                plus-petit(z1 y1) and
                                Volv(x) and
                                /

```

```

debit-aortique-cont-vent(x) if phase(contrac-ejec) and
                                Pvconnu(y) and
                                Psys(z) and
                                diffn(y z X) and
                                Raorte-pas(Y) and
                                Rcar(Z) and
                                addit(Y Z x1) and
                                div-PR(X x1 y1) and
                                /

```

### 7.11

2. Il est parfois nécessaire d'empêcher le backtracking lorsqu'une relation est définie par plusieurs règles de même conséquent et qu'une seule solution n'est admissible.

#### Exemple

Voir exemple en 1.

### 7.3 CRITIQUE DE LA PROGRAMMATION LOGIQUE ET DE MICRO-PROLOG

---

Les remarques que nous formulons ici proviennent essentiellement de l'expérience personnelle que nous avons faites de la programmation logique et de micro-Prolog en particulier. Nous ne dressons en aucun cas une liste complète des avantages et des inconvénients de la technique ou du langage.



### 7.3.1 CRITIQUE DE LA PROGRAMMATION LOGIQUE

Notre critique de la programmation logique se résume en une liste non exhaustive d'une série d'avantages et d'inconvénients observés dans le cadre de notre travail.

#### AVANTAGE

1. L'attrait d'un langage de programmation logique comme Prolog réside dans la facilité de programmer. Un programme Prolog peut être compris comme un ensemble de déclarations descriptives d'un problème. La sémantique déclarative de ce langage permet la compréhension d'un programme sans se préoccuper des détails d'exécution [37].

#### INCONVENIENTS

1. Le langage Prolog possède deux sémantiques, une sémantique déclarative et une sémantique procédurale ou opérationnelle qui décrit la séquence des états de l'exécution d'un programme. Il arrive malheureusement que la programmation qui semble simple au départ, se complexifie dans la mesure où l'utilisateur doit parfois tenir compte de l'aspect procédural du langage.

L'utilisateur énonce des propositions logiques sous la forme de clause de Horn :  $A \leftarrow B_1 \text{ and } B_2 \text{ and } \dots \text{ and } B_n$ . Ces clauses sont insérées dans la base de connaissances du système. Selon le principe de résolution par réfutation de Robinson, Prolog se chargera de vérifier si une proposition donnée est vraie ou fausse en fonction des clauses contenues dans la base.

En principe, l'ordonnancement des règles dans la base de connaissances est sans importance. En pratique, ce n'est malheureusement pas le cas. Il arrive parfois qu'il faille tenir compte de la manière dont fonctionne le système Prolog pour écrire les clauses dans un programme. Ainsi par exemple, si la base contient une séquence de règles de même prédicat, l'ordre d'apparition dans la séquence peut avoir une grande influence sur la cohérence des résultats (cfr. 7.2.5).



Un autre inconvénient du fonctionnement du système Prolog est que plus la base de connaissances contient de règles, plus il devient difficile au programmeur de corriger des erreurs ou de s'assurer que les résultats proposés par le système sont corrects.

2. Les temps d'exécution des programmes Prolog sont relativement longs. Cette lenteur provient à nouveau du mécanisme de fonctionnement du système et en l'occurrence du backtracking. Pour y remédier, il est conseillé de placer les règles les plus couramment utilisées au début de la base de connaissances et d'empêcher le backtracking chaque fois qu'il est possible de le faire en utilisant le prédicat "cut".
3. Le travail sur fichier est limité et assez complexe essentiellement en raison de son caractère séquentiel. En effet, les opérations d'ouverture, de lecture-écriture et de fermeture d'un fichier doivent être réalisées dans une même règle. Si l'une de ces trois opérations échoue qu'en est-il de l'état du fichier manipulé?

### 7.3.2 CRITIQUE DU LOGICIEL MICRO-PROLOG

Un des seuls avantages de l'outil que nous retenons est qu'il propose un utilitaire qui permet de construire des programmes dans une syntaxe plus simple que celle du Prolog pur. Malheureusement l'impression qui subsiste après avoir utilisé le logiciel est loin d'être positive. Nous citons ci-dessous les divers aspects négatifs que nous avons pus découvrir durant la phase d'implémentation.

#### 1. ESPACE DE TRAVAIL

Le logiciel micro-Prolog nécessite 128K RAM. Il occupe environ 96K et laisse 32K pour le système d'exploitation. Une fois chargé, micro-Prolog indique que l'espace mémoire disponible pour le stockage et l'exécution de programmes est de l'ordre de 62K. De plus, nous utilisons deux utilitaires (SIMPLE.LOG, EXREL.LOG) qui ne laissent à notre disposition que 45K d'espace de travail. Cet espace ne peut contenir que des bases de



connaissances relativement petites (environ 200 règles) et leur extension est fortement limitée.

Pour résoudre ce problème, micro-Prolog autorise l'utilisation de fichiers externes comme extension de l'espace de travail. Nous avons donc stocké sur disquettes tous les faits relatifs aux opérations qualitatives. Cependant, une conséquence directe de l'utilisation de ces fichiers externes est l'augmentation du temps de réponse qui est d'autant plus élevé que le nombre d'opérations qualitatives à effectuer est important.

## 2. UTILISATION DES FICHIERS

Micro-PROLOG n'admet pas plus de 4 fichiers ouverts en même temps. Ceci nous a contraintes à rassembler plusieurs opérations qualitatives dans le même fichier.

## 3. EDATEUR

L'éditeur micro-PROLOG est un éditeur orienté lignes. L'insertion et la correction des règles en sont rendues assez fastidieuses.

## 4. GESTION D'ECRAN

Micro-Prolog ne possède pas de gestion d'écran. Il devient alors très complexe de développer des interfaces utilisateurs agréables et conviviales.

## 5. MODULE

Selon le manuel d'utilisation, il est possible de créer des modules contenant des bases de connaissance différentes ou une partie d'une base de connaissance. L'option a été essayée mais chaque tentative s'est vue vouée à l'échec. S'agit-il d'une erreur de conception ou d'un manque d'information ?

## CHAPITRE 8

### COMPARAISON PROLOG-LISP

#### 8.1. POURQUOI UNE ETUDE COMPARATIVE ENTRE PROLOG ET LISP

---

Durant la phase de tests de notre système implémenté en Prolog, il s'est avéré que l'outil utilisé (micro-Prolog) ne donnait pas entière satisfaction. En effet, outre les problèmes déjà énoncés au point 7.3, il est apparu que le temps de réponse du système est inacceptable étant donné qu'il est de l'ordre de la minute. En raison de cet aspect négatif, nous nous sommes proposé de chercher un autre outil utilisable sur IBM-PC et de comparer ainsi les deux implémentations.

En outre, il nous a semblé intéressant d'étudier un autre langage couramment utilisé dans les applications d'intelligence artificielle afin de montrer que Prolog n'est pas le seul langage possible pour implémenter notre système.

Notre choix s'est arrêté sur le LISP et le logiciel IQ-LISP version 1.4.5 de 1983 (Integral Quality).

L'implémentation en Lisp s'est effectuée de manière analogue à celle en Prolog. Nous avons repris les connaissances du chapitre 4 et les choix relatifs au raisonnement qualitatif du chapitre 5 et nous avons construit notre programme Lisp.

Ce programme a été conçu uniquement dans un but de comparaison. Par conséquent, certains aspects du programme Prolog, non nécessaires à la comparaison, n'ont pas été repris. L'initialisation et certains artifices utilisés pour l'implémentation Prolog n'ont pas été programmés.

L'entièreté du programme Lisp, repris à l'annexe 4, ne sera pas considérée dans ce chapitre. Nous n'en citerons que les caractéristiques primordiales à l'étude des deux langages après



avoir énoncé brièvement les grands principes de chacun d'eux.

## 8.2. PRINCIPES DE BASE DU PROLOG ET DE LISP

---

Les langages Prolog et Lisp possèdent deux caractéristiques communes. Ils permettent, l'un et l'autre, le traitement de données symboliques et sont construits sur base de systèmes mathématiques. Prolog est un langage logique qui tient son origine dans la logique des prédicats du premier ordre tandis que Lisp est un langage fonctionnel créé à partir du calcul lambda de Church.

Un programme Prolog est constitué d'un ensemble de clauses de Horn (voir point 7.2). Cet ensemble est appelé base de connaissances. L'interrogation de cette base de connaissances se fait par l'entremise de requêtes. Le moteur d'inférence attaché au Prolog, basé sur le principe du "pattern matching" et de l'unification, se charge de résoudre le problème posé selon la méthode de résolution par réfutation. [14]

Un programme Lisp est construit à partir d'un ensemble de définition de fonctions. Chaque requête est une fonction à évaluer selon la valeur des arguments qui lui sont donnés. Le calcul du résultat d'une requête est effectué par une fonction Lisp prédéfinie appelée EVAL. EVAL recherche la définition de la fonction reprise dans la requête, lie les variables de la définition de la fonction aux arguments de la requête et évalue cette fonction sur base des valeurs de ces arguments. [36]

## 8.3. DIVERGENCES OBSERVEES ENTRE LISP ET PROLOG DANS LE CADRE DE NOTRE SYSTEME

---

Ce paragraphe met en lumière certaines caractéristiques divergentes des deux langages Lisp et Prolog ainsi que leur application au sein de notre système.

REMARQUE : Les clauses Prolog reprises dans les exemples cités ci-après sont construites selon la syntaxe simplifiée de micro-Prolog.

### 8.3.1. PROCEDURES PROLOG ET FONCTIONS LISP

Une procédure Prolog est définie comme l'ensemble des clauses ayant même prédicat.

Exemple :

Calcul du débit de la valve mitrale

```
débit-mit(x) if Pres(y) &
                  Pvconnu(z) &
                  plus-petit(y z) &
                  Rmitrale-bloc(x1) &
                  diffn(y z y1) &
                  div-PR(y1 x1 x)
débit-mit(x) if Pres(y) &
                  Pvconnu(z) &
                  plus-grand-eg(y z) &
                  Rmitrale-bloc(x1) &
                  diffn(y z y1) &
                  div-PR(y1 x1 x)
```

débit-mit est le prédicat de la procédure, débit-mit(x) est le conséquent des deux clauses et l'ensemble des expressions alignées après le if sont appelées les antécédents.

Chaque clause de la procédure décrit les conditions à vérifier pour que le conséquent de cette clause soit vrai. Remarquons qu'une clause peut avoir 0 antécédent. Nous appelons ces clauses des faits.

En Lisp, les procédures Prolog sont restreintes à de simples fonctions et les structures de données sont des listes. Une fonction Lisp a autant de branches de conditions qu'il y a de clauses dans une procédure Prolog.



Exemple :

```
(DEF 'DEBITMIT '(LAMBDA ()  
  (COND ((MITRALEOUV()) (DIV (DIFF (PRESV QV NV))  
                                RMITRALEPAS))  
    (T (DIV (DIFF (PRESV QV NV))  
              RMITRALEBLOC))))))
```

Grâce aux possibilités de test du Lisp (COND), la fonction évaluatrice est directement orientée vers les calculs à opérer pour trouver la solution de la fonction. En Prolog, toutes les clauses dont le prédicat "matche" celui du théorème à prouver sont essayées successivement jusqu'à ce qu'une clause, dont tous les antécédents puissent être vérifiés, soit trouvée. Dans de nombreux cas, les clauses d'une même procédure contiennent un même antécédent qui est évalué plusieurs fois au cours du processus de résolution.

Exemple :

Pour le calcul du débit passant par la valve mitrale :  
en Lisp, le test MITRALEOUV nous dirige directement vers le calcul à effectuer sans devoir tester deux fois l'état de la valve.

En Prolog, le moteur d'inférence essaie de prouver la première clause de la procédure en calculant successivement Pres(y), Pvconnu(z), plus-petit(y),... Si plus-petit(y z) échoue, la deuxième clause est essayée. Le moteur d'inférence recalcule Pres(y) et Pvconnu(z) qui ont déjà été calculés lors de l'évaluation de la première clause de la procédure.

A moins de construire un mécanisme d'enregistrement des clauses déjà évaluées, le moteur d'inférence Prolog effectue bon nombre d'opérations déjà exécutées précédemment.

### 8.3.2. BACKTRACKING PROLOG

En réponse à une question de l'utilisateur, Prolog exécute une tâche. Une question stipule qu'une conjonction de buts soit satisfaite. Prolog utilise les clauses connues dans sa base de connaissances pour satisfaire les buts concernés par la requête. Un fait peut permettre qu'un but soit satisfait immédiatement. Par contre, une règle ne peut que réduire la tâche initiale en une autre moins complexe qui consiste à satisfaire une conjonction de sous-buts. Si un but quelconque ne peut être



atteint, le mécanisme de backtracking est activé. Il consiste à reconsidérer ce qui a été fait pour tenter de satisfaire les buts par un autre chemin. Dans certains systèmes toutes les réponses possibles à une question sont recherchées automatiquement, dans d'autres cette recherche exige l'intervention de l'utilisateur.

Les deux inconvénients essentiels du backtracking sont le grand besoin de place mémoire et les temps d'exécution médiocres. Il permet toutefois la réalisation d'itérations de haut niveau. Cependant, dans la pratique courante, cette qualité du backtracking n'est pas toujours très appréciée surtout dans les cas où une seule solution est suffisante pour une requête.

Il existe éventuellement une possibilité d'empêcher le backtracking. Le programmeur peut introduire dans une clause le prédicat `cut` qui a pour effet de geler la partie de la clause qui précède ce prédicat. Aucune alternative n'est tentée pour satisfaire à nouveau les buts ou sous-buts qui précèdent le `cut`. Cependant, ce prédicat doit être utilisé avec la plus grande attention. Il arrive en effet, que le système se comporte d'une manière inattendue lorsque la politique d'utilisation des règles est modifiée. Une règle qui donne un résultat correct lorsqu'elle est utilisée d'une certaine façon, peut très bien produire une erreur lors d'une utilisation différente. [14]

### 8.3.3. OUTPUTS

Les mécanismes Prolog permettent, lors d'une résolution, d'obtenir la valeur de plusieurs variables inconnues à l'origine. Imaginons un système décrivant les liens familiaux de certains individus. La question suivante peut être posée au système : `which (x y : Parents (Toto x y))`. Cette question demande le nom des parents de Toto. La résolution de cette requête donne deux valeurs : une valeur instanciée à `x` et une autre instanciée à `y`.

Par contre, étant donné qu'une fonction n'admet qu'un et un seul résultat, l'évaluation d'une fonction Lisp ne donne jamais qu'une seule valeur en sortie.

Cette caractéristique intéressante de Prolog n'est guère mise à profit dans notre système car chaque clause est construite de telle manière qu'un seul résultat apparaisse en sortie.



Exemple :

```
which (x : débitmit (x))  
    résultat : une seule valeur représentant le débit  
                de la valve mitrale  
which (x : Pv (x))  
    (Pression du ventricule ?)  
    résultat : une seule valeur représentant la  
                pression dans le ventricule
```

#### 8.3.4. ARGUMENTS EN ENTREE ET EN SORTIE

Les arguments, en entrée et en sortie des clauses Prolog, ne doivent pas être différenciés à l'avance et peuvent varier d'un appel à l'autre. Ainsi, les requêtes sur un même prédicat peuvent avoir des significations différentes selon l'argument dont on spécifie la valeur, et les procédures Prolog peuvent être utilisées pour atteindre des buts distincts.

Exemple :

```
which (x y : Parents (Toto x y))  
    Cette requête demande le nom des parents de Toto  
which (x : Parents (x Jules Marie))  
    Cette requête demande le nom d'un enfant de Jules  
    et de Marie
```

Cette possibilité d'un multiple usage des procédures Prolog ne se retrouve pas en Lisp. Chaque argument explicité dans une définition d'une fonction est un argument en entrée et l'évaluation de la fonction donnera un seul résultat.

Encore une fois notre système n'utilise pas cette caractéristique Prolog. En effet, la grande majorité des clauses que nous avons écrites possèdent un seul argument qui est l'argument en sortie.

Exemple :

```
which (x : débitmit(x))  
    "débitmit" ne possède qu'un argument auquel sera  
    instanciée une valeur après résolution.  
which (x : Pv(x))  
    "Pv" ne possède qu'un argument auquel sera
```

instancié une valeur après résolution.

### 8.3.5. FORMALISATION DES CONNAISSANCES

Notre système comprend deux sortes de connaissances :

- 1) des équations mathématiques tirées du modèle électrique destinées au calcul des valeurs des variables du système,
- 2) des règles de raisonnement telles que la détermination de la position des valves, la détermination des phases, le choix du nombre d'élastiques autour du coeur et de la durée des intervalles lors de la propagation du temps,...

La formulation des connaissances de la première catégorie se rapproche davantage de la formulation de fonctions dans un langage comme Lisp que de la formulation des clauses d'un langage logique tel que Prolog. Ces équations sont des fonctions et il est plus naturel d'utiliser un langage fonctionnel pour les coder. En effet, l'équation  $P_v = V_{olv} * N_v$  peut directement se codifier en IQ-Lisp de la manière suivante :

```
(DEF 'PRESV '(LAMBDA (VOLV NV) (MULTV VOLV NV)))
```

Les connaissances de la deuxième catégorie n'ont pas ce caractère fonctionnel. La position des valves, la phase dans lequel se trouve le coeur,... sont déterminées lorsqu'un ensemble de conditions sont satisfaites. Nous retrouvons là le caractère logique et déclaratif de Prolog. Par conséquent, Lisp est moins bien adapté pour la formalisation de ce type de connaissance.

### 8.3.6. LISIBILITE

Les fonctions Lisp, de par leur taille et leur complexité, nuisent à la lisibilité d'un programme, tandis que Prolog permet à un programme d'être formulé en unités restreintes, chacune ayant une lecture déclarative naturelle.



### 8.3.7. TABLES ET CALCULS QUALITATIFS

#### TABLES QUALITATIVES

Chaque table qualitative est représentée en Prolog par un ensemble de faits avec même prédicat (voir figure 6.2). La base de faits contient autant de faits concernant le calcul qualitatif qu'il y a de paires d'arguments pour chaque table. Cette quantité impressionnante de faits utilise énormément de place mémoire, si bien que nous avons été obligées d'utiliser des fichiers externes pour les stocker (voir chapitre 6.2.1)

En Lisp, les tables sont représentées par autant de listes qu'il existe de tables qualitatives. Chaque liste est attachée à un atome dont le nom est représentatif de celui de la table. Ces listes sont formées de sous-listes dont le premier élément est la valeur du premier argument de l'opération et le deuxième élément est lui-même une liste. Cette liste est composée de sous-listes de deux éléments dont le premier est la valeur du deuxième argument et le deuxième est le résultat de l'opération pour les arguments concernés.

#### Exemple :

MULTVQE : atome auquel est liée la liste :

```
((O ((O O) (TP TP) (P TP) (MP P) (MG P) (G MP) (TG MP)))  
(TP ((O O) (TP TP) (P TP) (MP P) (MG MP) (G MG) (TG MG)))  
(P ((O TP) (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))  
(MP ((O TP) (TP P) (P P) (MP MP) (MG MG) (G G) (TG TG)))  
(MG ((O TP) (TP P) (P MP) (MP MP) (MG MG) (G G) (TG TG)))  
(G ((O P) (TP P) (P MP) (MP MP) (MG MG) (G TG) (TG TG)))  
(TG ((O P) (TP MP) (P MP) (MP MG) (MG G) (G TG) (TG TG))))
```

La formalisation des tables en Lisp semble beaucoup plus complexe mais la taille mémoire occupée en est grandement réduite.

#### CALCUL QUALITATIF

Le moteur d'inférence Prolog trouve la solution à un calcul qualitatif en passant en revue tous les faits jusqu'à trouver celui dont le prédicat et les deux premiers arguments "matchent" avec le prédicat et les deux premiers arguments de la



clause à prouver. En raison du nombre important de faits, cette recherche peut prendre un temps non négligeable.

La recherche d'une solution à un calcul qualitatif en Lisp est effectuée à l'aide d'une fonction définie pour chaque opération. Cette fonction est construite sur base des fonctions Lisp prédéfinies ASSOC et CADR. Elle donne la solution au calcul sans passer en revue tous les éléments de la liste.

Exemple 1 :

```
(DEF 'MULTV '(LAMBDA (X Y)
  (CADR (ASSOC Y (CADR (ASSOC X MULTVQE))))))
Où (ASSOC X Y) avec X : clé de recherche
      Y : liste composée de sous-listes
          de 2 éléments
donne la sous-liste de Y dont le premier
élément est égal à la clé X
```

```
Où (CADR X) avec X : liste
      est une composition des 2 fonctions CDR et CAR.
(CDR X) donne la liste X sans son premier
élément
(CAR X) donne le premier élément de la
liste X
```

Exemple 2 :

résolution d'un calcul qualitatif suivant la fonction et la liste présentées précédemment :

```
Soit (MULTV '0 'TP)

(ASSOC '0 MULTVQE) => (0 ((0 0) (TP TP) ... (TG MP)))
                    = L
(CADR L) => ((0 0) (TP TP) ... (TG MP))
          = L'
(ASSOC 'TP L') => (TP TP)

(CADR '(TP TP)) => TP
```

TP est le résultat de la multiplication de 0 par TP.

Notons que la représentation Lisp des tables qualitatives dépend des besoins du système (la fonction de recherche et les listes sont conçues en fonction du résultat à obtenir), tandis que celle prévue en Prolog est indépendante de toute



application.

#### 8.3.8. MISE A JOUR DE LA BASE DE CONNAISSANCES

La propagation du temps dans notre système suppose une mise à jour des certains faits de la base de connaissances comme le volume dans le ventricule, le volume dans l'aorte,... (voir chapitre 7.1). Cette mise à jour est réalisée en Prolog en deux opérations :

- 1) suppression du fait concerné
- 2) ajout d'un nouveau fait avec une nouvelle valeur d'argument

Exemple :

```
(DELCL (Volv MG))  
(ADDCL (Volv P))
```

En Lisp, un fait Prolog est représenté par un atome auquel est liée une valeur. Il suffit, lorsqu'une mise à jour est nécessaire, d'associer une autre valeur à l'atome avec l'instruction Lisp prédéfinie SETQ.

Exemple :

```
(SETQ Volv 'P)
```

#### 8.3.9. ORDONNANCEMENT DES CLAUSES

Nous avons vu au paragraphe 7.1.5 que l'ordre des clauses dans une procédure Prolog a de l'importance afin de fournir une solution correcte au problème posé. En Lisp, les procédures n'existent pas. Le programmeur définit ses fonctions sans se soucier de l'ordre dans lequel elles sont enregistrées.

#### 8.4. CONCLUSION

---

Prolog possède de nombreux avantages non négligeables sur Lisp, notamment la multiplicité d'usages des procédures grâce aux arguments en entrée et en sortie non fixés, la possibilité d'obtenir plusieurs valeurs en sortie, la recherche de toutes les solutions possibles grâce au backtracking, son aspect déclaratif... Pourtant, notre système est tel que la majorité de ces caractéristiques intéressantes du Prolog ne sont pas utilisées. En outre, il s'est avéré que le temps de réponse est catastrophique. Pour donner un ordre de grandeur, selon les essais que nous avons réalisés, le programme micro-Prolog (opérations qualitatives comprises) utilise 15 à 20 Kbytes en plus que le programme écrit en Lisp.

Le programme Lisp, quant à lui, occupe beaucoup moins de place mémoire. Par ailleurs, les tests exécutés ont montré que le temps de réponse du système Lisp est de l'ordre de la seconde. Le seul inconvénient qui nous paraît assez important est la piètre lisibilité d'un programme Lisp. Mais les fonctions de notre système ne sont jamais très complexes et restent, dans l'ensemble, assez compréhensibles.



## CHAPITRE 9

### CONCLUSION

Le système qui vous a été proposé est l'amorce d'un outil complet d'aide à l'enseignement des mécanismes de fonctionnement du coeur gauche humain. Ce système devait expliquer le comportement du coeur avec des termes analogues à ceux utilisés lors de cours ou de travaux pratiques donnés dans ce domaine. Comme un professeur de cardiologie exprime la valeur des différents composants du coeur de manière qualitative, notre système manipule, lui aussi, des expressions du même type. Son implémentation a été réalisée selon les techniques de la programmation logique et dans un langage couramment utilisé dans le domaine de l'intelligence artificielle : Prolog.

La réalisation de ce projet a nécessité une étape d'acquisition de connaissances propres au domaine cardiaque. Après avoir assisté à des séances de cours et après avoir longuement dialogué avec le professeur, nous étions en possession de deux types de connaissances. Les premières sont des équations mathématiques tirées d'un modèle électrique reproduisant le fonctionnement du ventricule gauche. Ces relations sont utilisées pour calculer la valeur des diverses variables représentant les éléments du coeur (pressions, volumes et débits). La deuxième classe d'informations se compose de règles de raisonnement employées par le professeur pour décrire et déterminer l'état du muscle cardiaque en fonction des valeurs de ses variables. Ces règles sont utilisées pour déterminer, par exemple, si une valve est fermée ou pas, si le ventricule éjecte ou pas, si le coeur est dans sa phase de relaxation, etc...

Pour satisfaire notre besoin d'exprimer les valeurs des variables en termes autres que numériques, nous avons dû choisir un ensemble de mots pouvant représenter qualitativement toutes les valeurs réelles du système physique. Ces termes et leur signification ont été déterminés en fonction de graphes représentant l'évolution des valeurs des pressions et des volumes au cours du cycle cardiaque. L'échelle ainsi obtenue comprend 7



grandeurs qualitatives : O, TP (très petit), P (petit), MP (moyennement petit), MG (moyennement grand), G (grand) et TG (très grand).

Il est à remarquer dès à présent que la signification des valeurs qualitatives dépend du contexte dans lequel elles sont exprimées. Le terme TP définissant la pression minimale dans l'aorte ne représente pas la même quantité que le terme TP utilisé pour exprimer une pression très basse dans le ventricule. En effet, la pression minimale dans l'aorte reste bien supérieure, pour des coeurs physiologiquement normaux, à la pression minimale dans le ventricule. Par conséquent, TP dans le contexte de l'aorte n'est pas égal à TP dans le contexte du ventricule. En raison de cette différence de signification au niveau des valeurs qualitatives, une distinction très nette doit se marquer entre le raisonnement tenu pour le ventricule et celui tenu pour l'aorte. Cette différence se concrétise par l'utilisation de deux contextes distincts lors de la détermination des valeurs des variables.

La valeur de chaque variable du coeur gauche est déterminée selon les équations du modèle électrique. Ces équations mettent en oeuvre les opérations suivantes : addition, soustraction, multiplication et division. En raison de l'utilisation de grandeurs qualitatives, il fut nécessaire de redéfinir ces opérations afin qu'elles puissent manipuler des mots plutôt que des nombres.

Une caractéristique inhérente au calcul qualitatif est, bien sûr, l'indétermination au niveau du résultat à fournir et ce, à cause de la signification assez floue de ses arguments. En effet, si nous additionnons les deux valeurs qualitatives G et TP, le résultat peut être la valeur G ou la valeur TG. Nous avons décidé de lever cette indétermination en fixant un seul résultat possible pour chaque calcul. La valeur choisie est celle qui a la plus grande possibilité de se produire et qui correspond le mieux au réel selon le contexte dans lequel il s'inscrit.

L'évolution du comportement cardiaque est réalisée selon un système temporel à deux instants : l'instant courant et l'instant suivant. Il est possible, à partir de la valeur du volume de sang dans le ventricule et du volume dans l'aorte, de déterminer, à un instant, la valeur de toutes les autres variables ainsi que l'état du système cardiaque. Par conséquent, le passage à



l'instant d'après est orchestrée par la seule propagation dans le temps des valeurs de ces volumes.

Enfin, l'implémentation de notre système a été effectuée en Prolog sur IBM-PC à l'aide du logiciel micro-Prolog. Comme cet outil ne nous satisfait pas pour de multiples raisons, entre autres le temps de réponse élevé, nous avons tenté de montrer qu'il est possible, voir avantageux, de l'implémenter dans un autre langage. Dès lors, nous avons établi une brève étude comparative entre Prolog et Lisp au vu de notre système en nous référant aux principales caractéristiques de notre système.

Le système résultant de notre étude est un outil d'aide à l'enseignement du coeur gauche fonctionnant selon un mécanisme de questions/réponses. L'étudiant peut poser toute une série de questions concernant le fonctionnement du coeur à un instant donné (valeurs des variables, état des valves,...). Il peut demander que le système se propage à l'instant suivant et étudier cet instant de manière analogue au précédent. Le système répond à toutes ces questions en termes qualitatifs sur base des informations contenues dans sa base de connaissances.

En outre, notre système est paramétrable en ce sens que l'étudiant choisit lui-même les valeurs des différents paramètres qui influencent le comportement cardiaque. Ainsi, il peut étudier les diverses conséquences d'une pathologie introduite dans le système par l'intermédiaire d'une valeur non physiologiquement normale donnée à un certain paramètre. Par exemple, il peut décider d'étudier le cas d'un coeur battant plus vite que la normale en diminuant la valeur du paramètre représentant la longueur du rythme cardiaque. Le choix de la pathologie à étudier doit se faire au moment de l'initialisation d'une simulation quand l'utilisateur introduit toutes les valeurs des paramètres. Il n'est, en aucun cas, possible de modifier une valeur en cours d'exécution.

Au sein des recherches effectuées à l'UCL en Woluwe sur l'enseignement supérieur assisté par ordinateur, notre système constitue, en quelque sorte, une innovation. En effet, les outils conçus précédemment par cette équipe de recherche manipulent exclusivement des données numériques. Notre système est le point de départ d'un projet très ambitieux. Ce projet consiste à créer un système complet capable à la fois de simuler qualitativement le fonctionnement du coeur et de la boucle circulatoire, de répondre à n'importe quelle question des étudiants, de justi-



fier et de commenter ses réponses de la manière la plus naturelle possible comme le ferait un professeur de physiologie. Notre mission était d'effectuer une première étude sur la simulation qualitative du coeur gauche et sur la représentation du raisonnement suivi par un professeur pour expliquer son fonctionnement. Ce travail peut être considéré comme un débroussaillage du projet initial servant à découvrir une manière de traiter le problème du raisonnement qualitatif.

Le résultat de cette étude n'est donc pas un système complet et opérationnel. Nous n'avons pas implémenté de mécanisme explicatif et nous ne nous sommes pas attardées très longtemps sur les aspects conviviaux et performants du système.

Il est possible de pallier à ces limites actuelles par différents travaux à réaliser dans le futur. La phase explicative, très importante pour la compréhension des divers mécanismes cardiaques, peut être créée de la façon suivante. Etant donné que notre système est implémenté en Prolog, il suffirait de mémoriser toutes les règles qui ont été utilisées pour atteindre le résultat fourni, toutes les instanciations qui ont été réalisées et de les imprimer au terminal selon les désirs de l'utilisateur. Les inconvénients majeurs de cette méthode sont la place mémoire occupée et le temps de réponse du système assez importants dus à l'utilisation du logiciel micro-Prolog. Il serait possible d'implémenter le tout à l'aide d'un autre outil Prolog plus performant. D'autre part, rien ne nous empêche de travailler avec un autre langage comme Lisp ainsi que nous l'avons fait remarquer au chapitre précédent. Mais si nous utilisons un tel langage, le mécanisme de justification risque d'être plus complexe à traiter.

La compréhension de l'étudiant peut également être améliorée par un couplage du système avec un outil graphique. Cet outil dessinerait, sur le même écran ou sur un deuxième, l'évolution des différentes valeurs des variables durant toute la simulation.

L'aspect convivial peut être amélioré par l'ajout d'une interface. Celle-ci pourrait s'occuper, d'une part, d'une gestion d'écran, d'autre part, d'une traduction en langage semi-naturel. L'utilisateur pourrait alors interroger le système sans se soucier de la syntaxe dans laquelle est écrit le programme, sans se soucier des noms exacts des variables. Le système pourrait renvoyer des réponses et des explications se rapprochant



davantage du langage naturel.

Le même travail peut être effectué pour la simulation et l'explication qualitative de la boucle circulatoire complète. Ce système physique étant beaucoup plus complexe que celui du coeur, il faut avant tout s'interroger sur la portabilité et l'efficacité de la méthode que nous avons suivie pour résoudre notre problème restreint.

Le système que nous avons conçu est loin d'être entièrement terminé, ses limites sont évidentes. Cependant, l'objectif que nous poursuivions n'était pas de concevoir un système optimal. Le prototype que nous avons élaboré constitue, en réalité, une première étude du raisonnement qualitatif adapté à la physiologie humaine.

## BIBLIOGRAPHIE

- [1] L. A. ZADEH  
A theory of approximate reasoning.  
Electronic research laboratory, University of California,  
August 1977
- [2] E. BALKOVICH, S. LERMAN, R. P. PARMELEE  
Computing in higher education : the athena experience.  
Communications of the ACM, Nov. 1985, Vol. 28
- [3] PATRICK H. McCANN  
Learning strategies and computer-based instruction.  
Computer & education, Vol. 5 p.133 to 140, 1981
- [4] J. LEFEVRE, J. SANDRAP, J. BARRETO  
A simple continuous system simulation package.  
HP 1000 Computer trends, ed. M. Beckman, A. van Putten,  
P. Luidema, Reedbooks, Survey England, April 1981
- [5] J. LEFEVRE, A. SONK, J. BARRETO  
Microcomputers and simulation in teaching of physiology.  
Archives Internationales de physiologie et biochimie, Vol.90,  
5, p.52-53, 1982
- [6] J. LEFEVRE, J. BARRETO  
Didactic microcomputer simulation in cardiac dynamics.  
5th Annual Conference on Frontiers of Engineering in Health  
Care (IEEE), Columbus, Ohio, USA, 1982
- [7] J. LEFEVRE, J. BARRETO, A. SONCK  
Teaching of cardiac dynamics by microcomputer simulation.  
IV Encontro Nacional de Automatica, Sociedade Brasileira de  
Automatica, julho 1983



- [8] J. LEFEVRE, J. BARRETO, H. POULEUR, G. PETIT  
Feasibility study of optimal control for a ventricular assist device.  
IV Encontro Nacional de Automatica, Sociedade Brasileira de Automatica, julho 1983
- [9] J. LEFEVRE, J. BARRETO, R. FABRI  
An authoring system for computer aided instruction and simulation in physiology.  
7th Annual conference of the IEEE Engineering in Medicine and Biology Society, p. 731-733, Chicago, Illinois, USA, sept 85
- [10] DANIEL G. BOBROW  
Qualitative reasoning about physical system : an introduction.  
Artificial intelligence, Vol. 24, 1984
- [11] J. DE KLEER, J. S. BROWN  
A qualitative physics based on confluences.  
Artificial intelligence, Vol. 24, 1984
- [12] K.D. FORBUS  
Qualitative process theory  
Artificial intelligence, Vol. 24, 1984
- [13] PAUL R. COHEN  
Heuristic reasoning about uncertainty : an artificial intelligence approach.  
Pitman advanced publishing program, 1985
- [14] W. F. CLOKSIN, C. S. MELLISH  
Programming in Prolog  
Springer-verlag, Berlin, Heidelberg, New-York, Tokyo  
1984
- [15] K. L. CLARK, F.G. McCABE  
micro-PROLOG : programming in logic.  
Prentice Hall International, 1984
- [16] VADOT  
Physiologie humaine, la circulation.  
Ed. Simep, Chap. 4, 5 et 6

- [17] ARTHUR C. GUYTON  
Textbook of medical physiology.  
Ed. W. B. Saunders Company, 1971
- [18] ROBERT F. RUSHMER  
Dynamique de l'appareil cardio-vasculaire.  
Ed. Boin
- [19] ALAN C. BURTON  
Physiology and biophysics of the circulation.  
Ed. Year book medical publishers, INC, 1975
- [20] J. BARRETO, M. NOIRHOMME-FRAITURE  
Minimal modeling : a bond graph approach.  
1986
- [21] E. BRENNER, M. JAVID  
Analysis of electric circuits.  
Mc Graw Hill, 1959
- [22] J. LEFEVRE, F. MESSEMACKERS, Y. RYCKMANS  
Un logiciel ouvert de simulation interactive dont le langage propre est le Fortran : son application à la modélisation des dyskinesies ventriculaires.  
Mémoire présenté en 1984, Louvain-en-Woluwe.
- [23] FR. HAYES-ROTH, DONALD A. WATERMAN, DOUGLAS B. LENAT  
Building experts systems.  
Addison-Wesley publishing Company, INC, 1983
- [24] FREDERICK HAYES-ROTH  
The knowledge-based expert system : a tutorial.  
Computer, INC, sept. 1984
- [25] E. CHOURAQUI, H. FARRENY, D. KAYER, H. PRADE  
Modelisation du raisonnement et de la connaissance.  
TSI, Techniques et sciences informatiques, Rapports
- [26] KEN FORBUS  
Measurement interpretation in qualitative process theory.  
Artificial intelligence, Vol. 24, 1984



- [27] BENJAMIN KUIPERS, JEROME P. KASSINER  
How to discover a knowledge representation for causal reasoning by studying an expert physician.  
Artificial intelligence, Vol. 24, 1984
- [28] PAUL R. COHEN, MILTON R. GRINSBERG  
A framework for heuristic reasoning about uncertainty.  
1982
- [29] HENRI PRADE  
A synthetic view of approximale reasoning technique.  
1982
- [30] EHUD Y. SHAPIRO  
Logic programs with uncertainties : a tool for implementing rule-based systems.  
1982
- [31] CLAUS-RAINER ROLLINGER  
How to represent evidence, aspects of uncertain reasoning.  
Technische Universitaet Berlin, Projekt KIT, Sekr, Fr 5-8,  
1982
- [32] KANDRASHINA E. Yu.  
Representation of temporel knowledge.  
AI Laboratory, Computing center, Siberian Branch of the USSR Academy of Sciences, Novosibirsk USSR, 1982
- [33] R. KOWALSKI  
Logic for problem solving.  
Computer science library, Artificial intelligence series,  
Nils J. NILSSON Ed.
- [34] JACQUES COHEN  
Describing Prolog by its interpretation and compilation.  
Communications of the ACM, Dec. 1985
- [35] F. G. McCABE, K. L. CLARK, B. D. STEEL  
micro-PROLOG 3.1, Programmeurs reference manual  
Logic Programming Associates ltd. 1984
- [36] H. WINSTON, B. HORN  
Lisp  
Addison-Wesley Publishing Company, Inc., 1981

- [36] CHRISTOPHER JOHN HOGGER  
Introduction to logic programming.  
Academic Press 1984
  
- [37] D. WARREN, L. PEREIRA  
Prolog - the language and its implementation compared with  
Lisp.  
Symposium on AI and programming languages, ACM, August 1977
  
- [38] E. TICK  
Memory performance of Lisp and Prolog programs.



FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX - NAMUR

INSTITUT D'INFORMATIQUE

CONTRIBUTION A LA CONCEPTION  
D'UN SYSTEME INFORMATIQUE D'AIDE A  
L'ENSEIGNEMENT DE LA PHYSIOLOGIE  
CARDIAQUE :

ELABORATION D'UN MODELE QUALITATIF  
DU COEUR GAUCHE SELON LES PRINCIPES  
DE LA PROGRAMMATION LOGIQUE

ANNEXES

ANNE HENRIOUL  
DOMINIQUE WARNIER

ANNEE ACADEMIQUE 1985-1986

MEMOIRE PRESENTE EN VUE DE  
L'OBTENTION DU GRADE DE LICENCIE  
ET MAITRE EN INFORMATIQUE

# ANNEXE 1

## TABLES DES OPERATIONS QUALITATIVES

### Calcul de la pression du ventricule      MULTV-QE

Volume du ventricule \* Elastance = Pression du ventricule

| $\begin{matrix} E \\ Q \end{matrix}$ | 0  | TP | P  | MP | MG | G  | TG |
|--------------------------------------|----|----|----|----|----|----|----|
| 0                                    | 0  | TP | TP | P  | P  | MP | MP |
| TP                                   | 0  | TP | TP | P  | MP | MG | MG |
| P                                    | TP | TP | P  | MP | MG | G  | TG |
| MP                                   | TP | P  | P  | MP | MG | G  | TG |
| MG                                   | TP | P  | MP | MP | MG | G  | TG |
| G                                    | P  | P  | MP | MP | MG | TG | TG |
| TG                                   | P  | MP | MP | MG | G  | TG | TG |



**Calcul de la pression de l'aorte****MULTAO-QE**

Volume de l'aorte \* Elastance - Pression de l'aorte

| <div>E<br/>Q</div> | 0  | TP | P  | MP | MG | G  | TG |
|--------------------|----|----|----|----|----|----|----|
| 0                  | 0  | 0  | TP | TP | TP | P  | P  |
| TP                 | 0  | TP | TP | P  | P  | P  | MP |
| P                  | 0  | TP | P  | P  | MP | MP | MG |
| MP                 | TP | P  | P  | MP | MG | MG | G  |
| MG                 | TP | P  | MP | MG | MG | G  | G  |
| G                  | P  | MP | MG | G  | G  | TG | TG |
| TG                 | P  | MP | MG | G  | TG | TG | TG |

**Calcul du volume du ventricule****DIVV-PE**

Pression du ventricule / Elastance = Volume du ventricule

| <div>P \ E</div> | 0  | TP | P  | MP | MG | G  | TG |
|------------------|----|----|----|----|----|----|----|
| 0                | TP | 0  | 0  | 0  | 0  | 0  | 0  |
| TP               | MP | MP | P  | P  | TP | TP | 0  |
| P                | G  | G  | MG | MP | P  | TP | TP |
| MP               | G  | G  | MG | MG | MP | P  | TP |
| MG               | TG | G  | G  | MG | MG | MP | P  |
| G                | TG | TG | G  | G  | MG | MP | MP |
| TG               | TG | TG | TG | G  | G  | MG | MP |



# **Calcul du volume de l'aorte      DIVAO-PE**

Pression de l'aorte / Elastance = Volume de l'aorte

| <div><div>P</div><div>E</div></div> | 0  | TP | P  | MP | MG | G  | TG |
|-------------------------------------|----|----|----|----|----|----|----|
| 0                                   | P  | TP | TP | 0  | 0  | 0  | 0  |
| TP                                  | MG | MP | MP | P  | P  | TP | 0  |
| P                                   | G  | G  | MG | MG | MP | P  | TP |
| MP                                  | TG | G  | G  | MG | MP | MP | P  |
| MG                                  | TG | TG | G  | G  | MG | MG | MP |
| G                                   | TG | TG | TG | G  | G  | MG | MP |
| TG                                  | TG | TG | TG | TG | G  | G  | MG |

**Calcul d'un flux :**

**Calcul de la difference des pressions      DIFFN**

Pression 1 - Pression 2

| P1 \ P2 | 0  | TP  | P   | MP  | MG  | G   | TG  |
|---------|----|-----|-----|-----|-----|-----|-----|
| 0       | 0  | -TP | -P  | -MP | -MG | -G  | -TG |
| TP      | TP | 0   | -TP | -P  | -MP | -MG | -G  |
| P       | P  | TP  | 0   | -TP | -P  | -MP | -MG |
| MP      | MP | P   | TP  | 0   | -TP | -P  | -MP |
| MG      | MG | MP  | P   | TP  | 0   | -TP | -P  |
| G       | G  | MG  | MP  | P   | TP  | 0   | -TP |
| TG      | TG | G   | MG  | MP  | P   | TP  | 0   |



**Calcul du flux****DIV-PR**

(Pres rés - Pres vent) / résistance mitrale

(Pres vent - Pres aorte) / résistance aorte

| $\begin{array}{c} R \\ \hline P1-P2 \end{array}$ | 0   | TP  | P   | MP  | MG  | G   | TG |
|--------------------------------------------------|-----|-----|-----|-----|-----|-----|----|
| 0                                                | 0   | 0   | 0   | 0   | 0   | 0   | 0  |
| TP                                               | TP  | TP  | TP  | TP  | TP  | TP  | 0  |
| P                                                | P   | P   | TP  | TP  | TP  | TP  | 0  |
| MP                                               | MP  | MP  | P   | TP  | TP  | TP  | 0  |
| MG                                               | MG  | MG  | MP  | P   | TP  | TP  | 0  |
| G                                                | G   | G   | MG  | MP  | P   | TP  | 0  |
| TG                                               | TG  | TG  | G   | MG  | MP  | P   | 0  |
| -TP                                              | -TP | -TP | -TP | -TP | -TP | -TP | 0  |
| -P                                               | -P  | -P  | -TP | -TP | -TP | -TP | 0  |
| -MP                                              | -MP | -MP | -P  | -TP | -TP | -TP | 0  |
| -MG                                              | -MG | -MG | -MP | -P  | -TP | -TP | 0  |
| -G                                               | -G  | -G  | -MG | -MP | -P  | -TP | 0  |
| -TG                                              | -TG | -TG | -G  | -MG | -MP | -P  | 0  |

**Additon de résistances      ADDIT**

| R1 \ R2 | 0  | TP | P  | MP | MG | G  | TG |
|---------|----|----|----|----|----|----|----|
| 0       | 0  | TP | P  | MP | MG | G  | TG |
| TP      | TP | TP | P  | MP | MG | G  | TG |
| P       | P  | P  | MP | MG | G  | TG | TG |
| MP      | MP | MP | MG | G  | G  | TG | TG |
| MG      | MG | MG | G  | G  | TG | TG | TG |
| G       | G  | G  | TG | TG | TG | TG | TG |
| TG      | TG | TG | TG | TG | TG | TG | TG |



# **Calcul du volume du ventricule à l'instant d'après      ADDITV**

Volume-vent( $t + \Delta t$ ) = Volume-vent( $t$ ) + variation-volume pendant  $\Delta t$

| $Q \backslash Iv * \Delta t$ | -TG | -G | -MG | -MP | -P | -TP | 0  | TP | P  | MP | MG | G  | TG |
|------------------------------|-----|----|-----|-----|----|-----|----|----|----|----|----|----|----|
| 0                            | 0   | 0  | 0   | 0   | 0  | 0   | 0  | TP | P  | MP | MG | G  | TG |
| TP                           | 0   | 0  | 0   | 0   | 0  | 0   | TP | TP | P  | MP | MG | G  | TG |
| P                            | 0   | 0  | 0   | 0   | TP | TP  | P  | P  | MP | MG | G  | G  | TG |
| MP                           | 0   | 0  | 0   | TP  | P  | P   | MP | MP | MG | G  | G  | TG | TG |
| MG                           | 0   | 0  | TP  | P   | P  | MP  | MG | MG | G  | G  | G  | TG | TG |
| G                            | 0   | TP | P   | P   | MP | MG  | G  | G  | G  | TG | TG | TG | TG |
| TG                           | TP  | P  | MP  | MP  | MG | G   | TG | TG | TG | TG | TG | TG | TG |

## **MULTV**

Variation de volume-vent =  $I_v * \Delta t$

| $\Delta t$<br>$I_v$ | TP  | P   | MP  | MG  | G   | TG  |
|---------------------|-----|-----|-----|-----|-----|-----|
| -TG                 | -TG | -TG | -TG | -TG | -TG | -TG |
| -G                  | -G  | -G  | -G  | -G  | -G  | -TG |
| -MG                 | -MG | -G  | -G  | -G  | -G  | -TG |
| -MP                 | -MP | -MG | -MG | -MG | -G  | -G  |
| -P                  | -P  | -MP | -MG | -MG | -G  | -G  |
| -TP                 | -TP | -P  | -MP | -MG | -G  | -G  |
| 0                   | 0   | 0   | 0   | 0   | 0   | 0   |
| TP                  | TP  | P   | MP  | MG  | G   | G   |
| P                   | P   | MP  | MG  | MG  | G   | G   |
| MP                  | MP  | MG  | MG  | MG  | G   | G   |
| MG                  | MG  | G   | G   | G   | G   | TG  |
| G                   | G   | G   | G   | G   | G   | TG  |
| TG                  | TG  | TG  | TG  | TG  | TG  | TG  |



# **Calcul du volume de l'aorte à l'instant d'après      ADDITAO**

Volume-aorte( $t + \Delta t$ ) = Volume-aorte( $t$ ) + variation-volume pendant  $\Delta t$

| Q \ Ia*Δt | -TG | -G | -MG | -MP | -P | -TP | 0  | TP | P  | MP | MG | G  | TG |
|-----------|-----|----|-----|-----|----|-----|----|----|----|----|----|----|----|
| 0         | 0   | 0  | 0   | 0   | 0  | 0   | 0  | TP | P  | MP | MG | G  | TG |
| TP        | 0   | 0  | 0   | 0   | 0  | TP  | TP | P  | MP | MG | MG | G  | TG |
| P         | 0   | 0  | 0   | 0   | TP | P   | P  | MP | MG | G  | G  | TG | TG |
| MP        | 0   | 0  | 0   | TP  | P  | MP  | MP | MG | G  | G  | G  | TG | TG |
| MG        | 0   | 0  | TP  | P   | MP | MG  | MG | MG | G  | G  | G  | TG | TG |
| G         | 0   | TP | P   | MP  | MG | G   | G  | G  | TG | TG | TG | TG | TG |
| TG        | TP  | P  | MP  | MG  | G  | TG  | TG | TG | TG | TG | TG | TG | TG |

## MULTAO

Variation de volume-aorte =  $I_a * \Delta t$

| $\Delta t$<br>$I_a$ | TP  | P   | MP  | MG  | G   | TG  |
|---------------------|-----|-----|-----|-----|-----|-----|
| -TG                 | -TG | -TG | -TG | -TG | -TG | -TG |
| -G                  | -G  | -G  | -G  | -TG | -TG | -TG |
| -MG                 | -MG | -MG | -MG | -G  | -G  | -G  |
| -MP                 | -MP | -MP | -MP | -MG | -MG | -G  |
| -P                  | -P  | -P  | -P  | -MP | -MP | -MP |
| -TP                 | -TP | -P  | -P  | -P  | -P  | -MP |
| 0                   | 0   | 0   | 0   | 0   | 0   | 0   |
| TP                  | TP  | P   | P   | P   | P   | MP  |
| P                   | P   | P   | P   | MP  | MP  | MP  |
| MP                  | MP  | MP  | MP  | MG  | MG  | G   |
| MG                  | MG  | MG  | MG  | G   | G   | G   |
| G                   | G   | G   | G   | TG  | TG  | TG  |
| TG                  | TG  | TG  | TG  | TG  | TG  | TG  |



## ANNEXE 2

### EXEMPLE D'EXECUTION DES PROGRAMMES PROLOG

#### (INITIALISATION ET PROGRAMME PRINCIPAL)

Le système micro-Prolog indique qu'il est prêt à recevoir une question en imprimant au terminal le prompt &.. Les questions posées en exemple par l'utilisateur sont donc les lignes précédées de &.. Les autres lignes sont les réponses fournies par le système.

&.is(initialisation)

INITIALISATION DES VALEURS DES PARAMETRES ET VARIABLES

-----

Chaque demande de valeur relative à un paramètre s'accompagne d'une liste des valeurs possibles parmi lesquelles vous devez effectuer votre choix, et de la valeur typiquement sélectionnée pour la simulation d'un coeur normal.

Si vous ne pouvez ou ne voulez pas choisir une valeur, tapez \$ en guise de réponse.

valeur de la pression du réservoir ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : P

votre choix : P

valeur de la résistance périphérique ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : MG

votre choix : MG

valeur de la résistance passante de la valve mitrale ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : 0

votre choix : 0

valeur de la résistance bloquante de la valve mitrale ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : TG

votre choix : TG

valeur de la résistance passante de la valve aortique ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : 0

votre choix : 0

valeur de la résistance bloquante de la valve aortique ?

valeurs possibles : 0 TP P MP MG G TG

valeur normale : TG

votre choix : TG



valeur de l'élastance maximale du ventricule ?

valeurs possibles : petit norm grand

valeur normale : norm

votre choix : norm

valeur du rythme cardiaque ?

valeurs possibles : court norm long

valeur normale : norm

votre choix : norm

valeur de la raideur de l'aorte ?

valeurs possibles : O TP P MP MG G TG

valeur normale : MP

votre choix : MP

valeur du volume dans le ventricule au début de la phase de repos?

valeurs possibles : O TP P MP MG G

valeur normale : TP

votre choix : TP

valeur du volume de l'aorte au début de la phase de repos ?

valeurs possibles : O TP P MP MG G TG

valeur normale : MG

votre choix : \$

valeur de la pression de l'aorte au début de la phase de repos ?

valeurs possibles : O TP P MP MG G TG

valeur normale : MP

votre choix : MP

YES

&.

&. is(help)

MENU

is(help-ventricule)

is(help-aorte)

is(help-debit)

is(help-phase)

is(help-parametres)

YES

&.

&. is(help-ventricule)

LISTE DES COMMANDES POSSIBLES :

is(vol-vent)

is(pres-vent)

is(debit-valve-mitrale)

is(debit-valve-aortique)

is(variation-vol-vent)

is(raidur-vent)

is(remplissage)

is(ejection)

YES

&. is(help-aorte)

LISTE DES COMMANDES POSSIBLES :

is(vol-aorte)

is(pres-aorte)

is(debit-aortique)

is(debit-reseau-periph)

is(variation-vol-aorte)

is(ejection)

YES

&.



```

&.is(help-debit)
LISTE DES COMMANDES POSSIBLES :
is(debit-valve-mitrale)
is(debit-valve-aortique)
is(debit-reseau-periph)
YES
&.

&.is(vol-vent)
TP
sens de variation : croissant
YES
&.is(pres-vent)
O
sens de variation : croissant
YES
&.is(debit-valve-mitrale)
P
YES
&.is(debit-valve-aortique)
contexte du ventricule : O
contexte de l'aorte : O
YES
&.is(debit-reseau-periph)
TP
YES
&.is(variation-vol-vent)
P
YES
&.is(variation-vol-aorte)
-TP
YES
&.is(phase-coeur)
phase : repos
YES
&.is(remplissage)
YES
&.is(ejection)
NO
&.

&.is(instant-apres)
phase : repos
YES
&.

```

&.is(vol-vent)

G

sens de variation : croissant

YES

&.is(pres-vent)

P

sens de variation : croissant

YES

&.

.....



# ANNEXE 3

## PROGRAMMES PROLOG

### 1. INITIALISATION

---

1. initialisation if entete &  
    texte-explic &  
    valeurs-par &  
    valeurs-var &  
    charg-prog

Clause principale à invoquer pour déclencher l'initialisation.

#### TEXTE EXPLICATIF

2. entete if PP &  
    PP (INITIALISATION DES VALEURS DE PARAMETRES ET  
        VARIABLES)

Clause faisant apparaître à l'écran un entête pour l'initialisation.

3. texte-explic if PP &  
    PP (Chaque demande relative à un paramètre s'accompagne d'une liste des valeurs possibles parmi lesquelles vous devez effectuer votre choix et de la valeur typiquement sélectionnée pour la simulation d'un coeur normal.) &  
    PP (Si vous ne pouvez ou voulez pas choisir une valeur, tapez \$ en guise de réponse.)

Impression au terminal d'un petit texte explicatif sur la manière d'introduire les valeurs.

## INTRODUCTION INTERACTIVE DES VALEURS DES PARAMETRES (ou VARIABLES)

Pour toutes valeurs de paramètres (variables) à introduire, il faut :

- a) Imprimer au terminal le nom du paramètre (variable),
- b) Imprimer au terminal la liste des valeurs possibles que peut prendre ce paramètre (variable),
- c) Imprimer au terminal la valeur normale pour ce paramètre (variable),
- d) Lire la valeur choisie par l'utilisateur,
- e) Introduire le fait correspondant dans la base de connaissances du système.

```
4. valeurs-par if val-Pres &
    val-Rper &
    val-Rmit-pas &
    val-Rmit-bloc &
    val-Raorte-pas &
    val-Raorte-bloc &
    val-Emax &
    val-raid-a &
    val-Rythme
```

```
5. val-Pres if PP &
    PP (Pression du réservoir ?) &
    PP (valeurs possibles : 0 TP P MP MG G TG) &
    PP (valeur normale : P) &
    P (votre choix :) &
    R (x) &
    IF ((EQ x $) ((add 1 (Pres)))
        ((add 1 (Pres(x)))))
```

```
6. val-Rmit-pas if
    PP &
    PP (Résistance passante de la valve mitrale ?) &
    PP (valeurs possibles : 0 TP P MP MG G TG) &
    PP (valeur normale : 0) &
    P (votre choix :) &
    R (x) &
    IF ((EQ x $) ((add 1 (Rmitrale-pas)))
        ((add 1 (Rmitrale-pas(x)))))
```



7. val-Rmit-bloc if

```

PP &
PP (Résistance bloquante de la valve mitrale ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : TG) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Rmitrale-bloc)))
      ((add 1 (Rmitrale-bloc(x)))))

```

8. val-Raorte-pas if

```

PP &
PP (Résistance passante da la valve aortique ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : 0) &

P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Raorte-pas)))
      ((add 1 (Raorte-pas(x)))))

```

9. val-Raorte-bloc if

```

PP &
PP (Résistance bloquante de la valve
    aortique ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : TG) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Raorte-bloc)))
      ((add 1 (Raorte-bloc(x)))))

```

10. val-Emax if PP &

```

PP (Elastance maximale du ventricule ?) &
PP (valeurs possibles : petit morn grand) &
PP (valeur normale : norm) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Emax)))
      ((add 1 (Emax(x)))))

```

11. val-Rythme if

```
PP &
PP (Rythme cardiaque ?) &
PP (valeurs possibles : court norm long) &
PP (valeur normale : norm) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Rythme)))
      ((add 1 (Rythme(x)))))
```

12. val-raïd-a if

```
PP &
PP (raïdeur de l'aorte ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : MP) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (raïd-a)))
      ((add 1 (raïd-a(x)))))
```

13. val-Rper if PP &

```
PP (Résistance du réseau phériphérique ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : MG) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Rper)))
      ((add 1 (Rper(x)))))
```

14. valeurs-var if val-vent &  
val-aorte

15. val-vent if PP &

```
PP (Volume dans ventricule au début de la phase
    de repos ?) &
PP (valeurs possibles : 0 TP P MP MG G) &
PP (valeur normale : TP) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((val-pres-vent))
      ((add 1 (Volv (x)))))
```

Si l'utilisateur connaît la valeur du volume de sang dans le ventricule, le système insère cette valeur dans la base de



connaissances, sinon le système demande la valeur de la pression dans le ventricule.

16. val-pres-vent if

```
PP &
PP (Pression du ventricule au début de la phase
    de repos ?) &
PP (valeurs possibles : 0 TP P) &
PP (valeur normale : 0) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((add 1 (Volv)))
      ((add 1 (Pvconnu(x) if /))
        (Volvent y)
        (add 1 (Volv (y))))))
```

Si l'utilisateur ne connaît pas la valeur de la pression dans le ventricule, le système insère un fait "Volv" sans argument dans la base de connaissances pour signifier que ni la valeur du volume dans le ventricule ni sa pression ne sont connues. Sinon le système insère la valeur de la pression dans la base de connaissances, calcul le volume correspondant et l'insère également dans la base de connaissances.

17. val-aorte if

```
PP &
PP (Volume de sang dans l'aorte au début de la
    phase de repos ?) &
PP (valeurs possibles : 0 TP P MP MG G TG) &
PP (valeur normale : MG) &
P (votre choix :) &
R (x) &
IF ((EQ x $) ((val-pres-aorte))
      ((add 1 (Volao(x))))))
```

Si l'utilisateur connaît la valeur du volume de sang dans l'aorte, le système insère cette valeur dans la base de connaissances, sinon le système demande la valeur de la pression dans l'aorte.

18. val-pres-aorte if

```
PP &  
PP (Pression dans l'aorte au début de la phase  
de repos ?) &  
PP (valeurs possibles : 0 TP P MP MG G TG) &  
PP (valeur normale : MP) &  
P (votre choix :) &  
R (x) &  
IF ((EQ x $) ((add 1 (Volao)))  
((add 1 (Paconnu(x) if /))  
(Volaorte y)  
(add 1 (Volao (y))))))
```

Si l'utilisateur ne connaît pas la valeur de la pression dans l'aorte, le système insère un fait "Volao" sans argument dans la base de connaissances pour signifier que ni la valeur du volume dans l'aorte ni sa pression ne sont connues. Sinon le système insère la valeur de la pression dans la base de connaissances, calcul le volume correspondant et l'insère également dans la base de connaissances.

19. Volvent (x) if Pvconnu(y) &  
divv-PE (y 0 x)

Calcul de la valeur du volume de sang dans le ventricule.

20. Volaorte (x) if Paconnu(y) &  
divao-PE (y 0 x)

Calcul de la valeur du volume de sang dans l'aorte.

21. divv-PE x if RPRED (B:PRESVOL (17 5) (25 16) (divv-PE x))

Opération nécessaire au calcul du volume dans le ventricule.

22. divao-PE x if RPRED (B:PRESVOL (25 16) (33 57) (divao-PE x))

Opération nécessaire au calcul du volume dans l'aorte.



## CHARGEMENT DU PROGRAMME PRINCIPAL

```
23. charg-prog if kill (entete) &  
      kill(texte-explic) &  
      kill (valeurs-var) &  
      kill (valeurs-par) &  
      kill (val-Rper) &  
      kill (val-Pres) &  
      kill (val-Rmit-pas) &  
      kill (val-Rmit-bloc) &  
      kill (val-Raorte-pas) &  
      kill (val-Raorte-bloc) &  
      kill (val-Emax) &  
      kill (val-Rythme) &  
      kill (val-raid-a) &  
      kill (val-vent) &  
      kill (val-pres-vent) &  
      kill (val-aorte) &  
      kill (val-pres-aorte) &  
      load (b:COEUR)
```

Destruction de toutes les clauses qui ont servi à  
l'initialisation. Chargement du programme principal.

## 2. PROGRAMME PRINCIPAL

---

- |     |                   |   |                                                  |
|-----|-------------------|---|--------------------------------------------------|
| 1.  | Pres(P)           | : |                                                  |
| 2.  | raid-a(MP)        | : |                                                  |
| 3.  | Psys(0)           | : |                                                  |
| 4.  | Rper(MG)          | : |                                                  |
| 5.  | Rmitrale-pas(0)   | : |                                                  |
| 6.  | Rmitrale-bloc(TG) | : | Liste des faits indiquant la valeur              |
| 7.  | Raorte-pas(0)     | : | donnée aux paramètres du système.                |
| 8.  | Raorte-bloc(TG)   | : |                                                  |
| 9.  | Rcar(TP)          | : |                                                  |
| 10. | Rythme(norm)      | : |                                                  |
| 11. | Emax(norm)        | : |                                                  |
| 12. | AugmNv(0)         | : |                                                  |
| 13. | Nv(0)             | : |                                                  |
| 14. | repos(debut)      | : |                                                  |
| 15. | ejec(non)         | : |                                                  |
|     |                   |   |                                                  |
| 16. | multv-QE          | : |                                                  |
| 17. | divv-PE           | : |                                                  |
| 18. | multao-QE         | : | Liste des opérations qualitatives définies.      |
| 19. | divao-PE          | : | Ces opérations sont stockées sur disquettes      |
| 20. | div-PR            | : | dans des fichiers externes micro-Prolog. Toute   |
| 21. | addit             | : | référence à l'une d'elles, en cours d'exécution, |
| 22. | diffn             | : | implique une lecture sur disque.                 |
| 23. | additv            | : |                                                  |
| 24. | multv             | : |                                                  |
| 25. | additao           | : |                                                  |
| 26. | multao            | : |                                                  |
|     |                   |   |                                                  |
| 27. | Pvconnu(X) if     |   |                                                  |
|     | Pv(X)             |   |                                                  |

Artifice de programmation, dans la mesure où les règles permettant de calculer les débits et les flux de sang dans le système utilisent une presssion du ventricule connue Pvconnu.



28. Pv(X) if  
     Volv(Y) and  
     Nv(Z) and  
     multv-QE(Y Z X) and  
     /

Lorsque le volume de sang dans le ventricule et l'élastance du ventricule sont connus, il est possible de déterminer la pression du ventricule par le produit de ces éléments.

29. Volv(TP)

Le volume de sang dans le ventricule est connu et est égal à TP

30. Paconnu(X) if  
     Pa(X)

Artifice de programmation, dans la mesure où les règles permettant de calculer les débits et les flux de sang dans le système utilisent une pression de l'aorte connue Paconnu.

31. Pa(X) if  
     ejec(oui) and  
     Pv(Y) and  
     Pv-cont-aorte(X) and  
     /

Lorsque le système est en phase d'éjection, la pression dans l'aorte est déterminée par la pression du ventricule à laquelle elle est égale.

32. Pa(X) if  
     Volao(Y) and  
     raid-a(Z) and  
     multao-QE(Y Z X) and  
     /

Lorsque le système n'est pas en phase d'éjection, que le volume de sang dans l'aorte et la raideur de l'aorte sont connus (Volao, raid-a), il est possible de déterminer la pression dans l'aorte (Pa) par le produit de ces éléments.

33. Volao(MG)

Le volume de sang dans l'aorte est connu et est égal à MG

34. vol-vent if

Volv(X) and  
evolutionVolv(Y) and  
PP(X) and  
PP(sens de variation : Y) and  
/

Règle permettant à l'utilisateur de demander le volume de sang dans le ventricule (Volv) et le sens d'évolution de ce volume (evolutionVolv). Cette règle permet de déterminer le volume sur base du volume du ventricule déjà connu.

35. vol-vent if

Pvconnu(X) and  
Nv(Y) and  
divv-PE(X Y Z) and  
evolutionPv(x) and  
PP(Z) and  
PP(sens de variation : x) and  
/

Règle permettant à l'utilisateur de demander le volume de sang dans le ventricule (Volv) et le sens d'évolution de ce volume (evolutionVolv). Cette règle permet de déterminer le volume sur base de la pression régnant dans le ventricule et de son élastance.

36. vol-aorte if

Volao(X) and  
evolutionVolao(Y) and  
PP(X) and  
PP(sens de variation : Y) and  
/

Règle permettant à l'utilisateur de demander le volume de sang dans l'aorte (Volao) et le sens de variation de ce volume (evolutionVolao). Cette règle permet de déterminer le volume sur base du volume de l'aorte déjà connu.



37. vol-aorte if

Paconnu(X) and  
raid-a(Y) and  
divao-PE(X Y Z) and  
evolutionPa(x) and  
PP(Z) and  
PP(sens de variation : x) and  
/

Règle permettant à l'utilisateur de demander le volume de sang dans l'aorte (Volao) et le sens de variation de ce volume (evolutionVolao). Cette règle permet de déterminer le volume sur base de la pression régnant dans l'aorte et de sa raideur.

38. pres-vent if

Pv(X) and  
evolutionPv(Y) and  
PP(X) and  
PP(sens de variation : Y) and  
/

Règle permettant à l'utilisateur de demander la pression dans le ventricule ainsi que le sens de variation de celle-ci.

39. pres-aorte if

Pa(X) and  
evolutionPa(Y) and  
PP(X) and  
PP(sens de variation : Y) and  
/

Règle permettant à l'utilisateur de demander la pression dans l'aorte ainsi que le sens de variation de celle-ci.

40. evolutionVolv(croissant) if

phase(repos)

Le volume de sang dans le ventricule gauche croît, si le système est en phase de repos.

41. evolutionVolv(constant) if  
    phase(contrac-iso)

Le volume de sang dans le ventricule gauche reste constant, si le système est en phase de contraction isovolumétrique.

42. evolutionVolv(decroissant) if  
    phase(contrac-ejec)

Le volume de sang dans le ventricule gauche décroît, si le système est en phase de contraction éjection.

43. evolutionVolv(constant) if  
    mitrale-fermee and  
    phase(relaxation)

Le volume de sang dans le ventricule gauche reste constant, si le système est en phase de relaxation et que la valve mitrale est fermée.

44. evolutionVolv(croissant) if  
    mitrale-ouverte and  
    phase(relaxation)

Le volume de sang dans le ventricule gauche croît, si le système est en phase de relaxation et que la valve mitrale est ouverte.

45. evolutionPv(croissant) if  
    phase(repos)

La pression au sein du ventricule croît, si le système est en phase de repos.

46. evolutionPv(croissant) if  
    phase(contrac-iso)

La pression au sein du ventricule croît, si le système est en phase de contraction isovolumétrique.



47. evolutionPv(croissant) if  
    phase(contrac-ejec) and  
    evolutionPa(croissant)

La pression au sein du ventricule croît, si le système est en phase de contraction éjection et que la pression au sein de l'aorte croît.

48. evolutionPv(decroissant) if  
    phase(contrac-ejec) and  
    evolutionPa(decroissant)

La pression au sein du ventricule décroît, si le système est en phase de contraction éjection et que la pression au sein de l'aorte décroît.

49. evolutionPv(decroissant) if  
    phase(relaxation)

La pression au sein du ventricule décroît, si le système est en phase de relaxation.

50. evolutionVolao(decroissant) if  
    phase(repos)

Le volume de l'aorte décroît si, le système est en phase de repos.

51. evolutionVolao(decroissant) if  
    phase(contrac-iso)

Le volume de l'aorte décroît, si le système est en phase de contraction isovolumétrique.

52. evolutionVolao(croissant) if  
    phase(contrac-ejec) and  
    var-Volao(X) and  
    pos(X)

Le volume de sang dans l'aorte croît, si le système est en phase de contraction éjection et si la variation de volume de sang dans l'aorte (var-Volao) est positive.

53. evolutionVolao(decroissant) if  
    phase(contrac-ejec) and  
    var-Volao(X) and  
    not pos(X)

Le volume de sang dans l'aorte décroît, si le système est en phase de contraction éjection et si la variation de volume dans l'aorte (var-Volao) est négative ou nulle.

54. evolutionVolao(decroissant) if  
    phase(relaxation)

Le volume de sang dans l'aorte décroît, si le système est en phase de relaxation.

55. evolutionPa(decroissant) if  
    phase(repos)

La pression au sein de l'aorte décroît, si le système est en phase de repos.

56. evolutionPa(decroissant) if  
    phase(contrac-iso)

La pression au sein de l'aorte décroît, si le système est en phase de contraction isovolumétrique.

57. evolutionPa(croissant) if  
    phase(contrac-ejec) and  
    evolutionVolao(croissant)

La pression au sein de l'aorte croît, si le système est en phase de contraction éjection et si le volume de sang dans l'aorte croît.

58. evolutionPa(decroissant) if  
    phase(contrac-ejec) and  
    evolutionVolao(decroissant)

La pression au sein de l'aorte décroît, si le système est en phase de contraction éjection et si le volume de sang dans l'aorte décroît.



59. evolutionPa(decroissant) if  
    phase(relaxation)

La pression au sein de l'aorte décroît si le système est en phase de relaxation.

60. remplissage if  
    evolutionVolv(croissant)

Le ventricule gauche se remplit si le volume de sang dans le ventricule croît.

61. ejection if  
    evolutionVolv(decroissant)

Le ventricule gauche éjecte du sang dans l'aorte, si le volume de sang dans le ventricule décroît.

62. debit-valve-aortique if  
    debit-aortique-cont-vent(X) and  
    debit-aortique-cont-aorte(Y) and  
    PP(contexte du ventricule : X) and  
    PP(contexte de l'aorte : Y)

Règle permettant à l'utilisateur de demander le débit de la valve aortique.

63. debit-valve-mitrale if  
    debit-mit(X) and  
    PP(X)

Règle permettant à l'utilisateur de demander le débit de la valve mitrale.

64. debit-reseau-periph if  
    debit-per(X) and  
    PP(X)

Règle permettant à l'utilisateur de demander le débit vers le réseau périphérique.

65. variation-vol-vent if  
var-Volv(X) and  
PP(X)

Règle permettant à l'utilisateur de demander la variation de volume de sang dans le ventricule.

66. variation-vol-aorte if  
var-Volao(X) and  
PP(X)

Règle permettant à l'utilisateur de demander la variation de volume de sang dans l'aorte.

67. debit-mit(X) if  
Pres(Y) and  
Pvconnu(Z) and  
plus-grand-eg(Y Z) and  
Rmitrale-pas(x) and  
diffn(Y Z y) and  
div-PR(y x X)

Le débit de la valve mitrale

pression réservoir - pression ventricule  
-----  
résistance passante de la valve mitrale

Si la pression dans le réservoir est supérieure ou égale à la pression dans le ventricule.



68. debit-mit(X) if  
 Pres(Y) and  
 Pvconnu(Z) and  
 plus-petit(Y Z) and  
 Rmitrale-bloc(x) and  
 diffn(Y Z y) and  
 div-PR(y x X)

Le débit de la valve mitrale

pression réservoir - pression ventricule  
 -----  
 résistance bloquante

Si la pression dans le réservoir est strictement inférieure à la pression dans le ventricule.

69. debit-aortique-cont-vent(X) if  
 Pvconnu(Y) and  
 Pa-cont-vent(x) and  
 plus-petit(Y x) and  
 Raorte-bloc(y) and  
 diffn(Y x z) and  
 div-PR(z y X) and  
 /

Le débit de la valve aortique dans le contexte du ventricule

pression ventricule - pression aorte contexte ventricule  
 -----  
 résistance passante de la valve aortique

Si la pression dans le ventricule est strictement inférieure à la pression aortique dans le contexte du ventricule.

```

70. debit-aortique-cont-vent(X) if
    phase(contrac-ejec) and
    Pvconnu(Y) and
    Psys(Z) and
    diffn(Y Z x) and
    Raorte-pas(y) and
    Rcar(z) and
    addit(y z X1) and
    div-PR(x X1 Y1) and
    Volv(Z1) and
    plus-petit(Z1 Y1) and
    Volv(X) and
    /

```

Le débit de la valve aortique dans le contexte du ventricule est égal au volume de sang résiduel du ventricule

Si le système est en phase d'éjection

ET

Si le quotient

$$\frac{\text{pression ventricule} - \text{pression périphérique}}{\text{résistance passante valve ao} + \text{résistance caractéristique}}$$

est strictement supérieur au volume de sang résiduel du ventricule.



```

71. debit-aortique-cont-vent(X) if
    phase(contrac-ejec) and
    Pvconnu(Y) and
    Psys(Z) and
    diffn(Y Z x) and
    Raorte-pas(y) and
    Rcar(z) and
    addit(y z X1) and
    div-PR(x X1 X) and
    /

```

Le débit de la valve aortique dans le contexte du ventricule

pression ventricule - pression périphérique  
 -----  
 résistance passante valve ao - résistance caractéristique

Si le système est en phase d'éjection

ET

Si le volume de sang résiduel du ventricule est supérieur ou égal au débit calculé (condition implicite de part l'ordonnement des règles).

```

72. debit-aortique-cont-aorte(X) if
    Pv-cont-aorte(Z) and
    Paconnu(x) and
    plus-petit(Z x) and
    Raorte-bloc(y) and
    diffn(Z x z) and
    div-PR(z y X) and
    /

```

Le débit de la valve aortique dans le contexte de l'aorte

pression ventricule contexte aorte - pression aorte  
 -----  
 résistance bloquante de la valve aortique

Si la pression ventriculaire dans le contexte de l'aorte est strictement inférieure à la pression dans l'aorte.

```

73. debit-aortique-cont-aorte(X) if
    phase(contrac-ejec) and
    Pv-cont-aorte(Z) and
    Psys(x) and
    diffn(Z x y) and
    Raorte-pas(z) and
    Rcar(X1) and
    addit(z X1 Y1) and
    div-PR(y Y1 Z1) and
    Volv(x1) and
    plus-petit(x1 Z1) and
    Volv(X) and
    /

```

Le débit de la valve aortique dans le contexte de l'aorte est égal au volume de sang résiduel du ventricule

Si le système est en phase d'éjection

ET

Si le quotient

$$\frac{\text{pression ventricule contexte aorte} - \text{pression aorte}}{\text{résistance passante valve ao} - \text{résistance caractéristique}}$$

est strictement supérieur au volume de sang résiduel du ventricule.



```

74. debit-aortique-cont-aorte(X) if
    phase(contrac-ejec) and
    Pv-cont-aorte(Z) and
    Psys(x) and
    diffn(Z x y) and
    Raorte-pas(z) and
    Rcar(X1) and
    addit(z X1 Y1) and
    div-PR(y Y1 X) and
    /

```

Le débit de la valve aortique dans le contexte de l'aorte

pression ventricule contexte aorte - pression périphérique  
 -----  
 résistance passante valve ao - résistance caractéristique

Si le système est en phase d'éjection

ET

Si le volume de sang résiduel du ventricule est supérieur ou égal au débit calculé (condition implicite de part l'ordonnement des règles).

```

75. debit-per(X) if
    Paconnu(Y) and
    Rper(Z) and
    div-PR(Y Z X)

```

Le débit vers le réseau périphérique

pression aorte - pression périphérique  
 -----  
 résistance périphérique

or la pression du réseau périphérique est considérée comme négligeable c'est pourquoi le

Le débit vers le réseau périphérique

pression aorte  
 -----  
 résistance périphérique

76. var-Volv(X) if  
     debit-mit(Y) and  
     debit-aortique-cont-vent(Z) and  
     diffn(Y Z X)

La variation du volume de sang dans le ventricule (var-Volv) est obtenue par la différence entre le débit de la valve mitrale (debit-mit) et le débit de la valve aortique dans le contexte du ventricule (debit-aortique-cont-vent).

77. var-Volao(X) if  
     debit-aortique-cont-aorte(Y) and  
     debit-per(Z) and  
     diffn(Y Z X)

La variation du volume de sang dans l'aorte (var-Volao) est obtenue par la différence entre le débit de la valve aortique dans le contexte de l'aorte (debit-aortique-cont-aorte) et le débit vers le réseau périphérique (debit-per).

78. phase-coeur if phase(X) and  
     PP(phase : X) and  
     /

Règle permettant à l'utilisateur de demander la phase dans laquelle se trouve le coeur gauche.

79. phase(repos) if  
     AugmNv(0) and  
     /

Le système est en phase de repos si le nombre d'élastiques autour du coeur est constant (AugmNv(0)).

80. phase(contrac-iso) if  
     aortique-fermee and  
     activite-croiss and  
     /

Le système est en phase de contraction isovolumétrique si la valve aortique est fermée (aortique-fermee) et si le coeur gauche est en activité croissante (activite-croiss).



81. phase(contrac-ejec) if  
    aortique-ouverte and  
    activite-croiss and  
    /

Le système est en phase de contraction éjection si la valve aortique est ouverte (aortique-ouverte) et si le ventricule gauche est en activité croissante (activite-croiss).

82. phase(relaxation) if  
    activite-decroiss

Le système est en phase de relaxation si le ventricule gauche est en activité décroissante (activite-decroiss).

83. mitrale-fermee if  
    Pvconnu(X) and  
    Pres(Y) and  
    plus-petit(Y X)

La valve mitrale est fermée si la pression du réservoir est strictement inférieure à la pression du ventricule.

84. mitrale-ouverte if  
    Pvconnu(X) and  
    Pres(Y) and  
    plus-grand-eg(Y X)

La valve mitrale est ouverte si la pression du réservoir est supérieure ou égale à la pression du ventricule.

85. aortique-fermee if  
    Pvconnu(X) and  
    Pa-cont-vent(Z) and  
    plus-petit(X Z)

La valve aortique est fermée si la pression du ventricule est strictement inférieure à la pression de l'aorte dans le contexte général qui est celui du ventricule.

```

86. aortique-ouverte if
    ejec(non) and
    Pvconnu(X) and
    Pa-cont-vent(Z) and
    plus-grand-eg(X Z) and
    DELCL(((ejec non))) and
    ADDCL(((ejec oui))) and
    /

```

La valve aortique est ouverte si le système n'est pas en phase d'éjection et si la pression dans le ventricule est supérieure ou égale à la pression aortique dans le contexte général.

NB: -Cette règle a également pour effet de supprimer un fait indiquant que le système n'est pas en phase d'éjection et d'ajouter un autre fait indiquant que le système passe en phase d'éjection puisque la valve aortique est ouverte. Le nouveau fait introduit permet de calculer la pression dans l'aorte sur base de celle régnant dans le ventricule lorsque le système est en phase d'éjection.

-Cette règle sera vérifiée lorsque le système entrera pour la première fois en phase d'éjection.

```

87. aortique-ouverte if
    ejec(oui)

```

La valve aortique est ouverte si le système est en phase d'éjection mais plus pour la première fois.

NB: Cette règle sera vérifiée le système est en phase d'éjection mais plus pour la première fois.

```

88. activite-croiss if
    AugmNv(+)

```

Le coeur gauche est en activité croissante si le nombre d'élastiques autour du coeur augmente (AugmNv(+)).

```

89. activite-decroiss if
    AugmNv(-)

```

Le coeur gauche est en activité décroissante si le nombre d'élastiques autour du coeur diminue (AugmNv(-)).



90. instant-apres if  
     Volv-apres(X) and  
     Volao-apres(Z) and  
     Nv-apres(Z) and  
     DELCL(((Volv x))) and  
     ADDCL(((Volv X))) and  
     DELCL(((Volao y))) and  
     ADDCL(((Volao Y))) and  
     phase(z) and  
     PP(phase : z)

Le système est propagé dans le temps, à l'instant  $(t + \Delta t)$ , s'il est possible de propager le volume de sang du ventricule à l'instant  $(t + \Delta t)$ ; s'il est possible de propager le volume de sang de l'aorte à l'instant  $(t + \Delta t)$ ; s'il est possible de propager le nombre d'élastiques autour du coeur à l'instant  $(t + \Delta t)$ ; s'il est possible de supprimer les faits concernant les valeurs des volumes de l'instant  $t$ ; s'il est possible d'ajouter les faits concernant les valeurs des volumes à l'instant  $(t + \Delta t)$ ; s'il est possible de déterminer la phase dans laquelle se trouve le système après propagation du temps.

91. Volv-apres(X) if  
     Volv(Y) and  
     var-Volv(Z) and  
     dt(x) and  
     multv(Z x y) and  
     additv(Y y X)

Le volume de sang dans le ventricule à l'instant  $(t + \Delta t)$  (Volv-apres) est égal au volume de sang dans le ventricule à l'instant  $t$  (Volv) augmenté du produit de la dérivée du volume de sang dans le ventricule à l'instant  $t$  (var-Volv) par l'intervalle de temps  $\Delta t$  (dt).

92. Volao-apres(X) if  
     Volao(Y) and  
     var-Volao(Z) and  
     dt(x) and  
     multao(Z x y) and  
     additao(Y y X)

Le volume de sang dans l'aorte à l'instant  $(t + \Delta t)$  (Volao-apres) est égal au volume de sang dans l'aorte à l'instant  $t$  (Volao) augmenté du produit de la dérivée du volume de sang dans l'aorte à l'instant  $t$  (var-Volao) par l'intervalle de temps  $\Delta t$  (dt).

93. raideur-vent if Nv(X) and  
     PP(raideur ventricule : X) and  
     /

Règle permettant à l'utilisateur de demander la raideur du ventricule gauche.

94. Nv-apres(0) if  
     repos(debut) and  
     Nv(0) and  
     DELCL(((repos debut))) and  
     ADDCL(((repos fin)))

Voir remarque 1.

95. Nv-apres(0) if  
     AugmNv(-) and  
     Nv(P) and  
     Emax(norm) and  
     DELCL(((Nv P))) and  
     DELCL(((AugmNv -))) and  
     DELCL(((repos))) and  
     ADDCL(((Nv 0))) and  
     ADDCL(((AugmNv 0))) and  
     ADDCL(((repos debut)))

Voir remarque 1.



95. Nv-apres(P) if  
 Nv(0) and  
 repos(fin) and  
 Emax(norm) and  
 DELCL(((repos fin))) and  
 DELCL(((Nv 0))) and  
 DELCL(((AugmNv 0))) and  
 ADDCL(((repos))) and  
 ADDCL(((Nv P))) and  
 ADDCL(((AugmNv +)))

Voir remarque 1.

96. Nv-apres(MP) if  
 AugmNv(+) and  
 Nv(P) and  
 Emax(norm) and  
 DELCL(((Nv P))) and  
 ADDCL(((Nv MP)))

Voir remarque 1.

97. Nv-apres(MG) if  
 AugmNv(+) and  
 Nv(MP) and  
 Emax(norm) and  
 DELCL(((Nv MP))) and  
 ADDCL(((Nv MG)))

Voir remarque 1.

98. Nv-apres(G) if  
 AugmNv(+) and  
 Nv(MG) and  
 Emax(norm) and  
 DELCL(((Nv MG))) and  
 ADDCL(((Nv G))) and  
 augmRao

Voir remarque 1.

99. Nv-apres(MG) if  
 AugmNv(+) and  
 Nv(G) and  
 Emax(norm) and  
 DELCL(((AugmNv +))) and  
 DELCL(((Nv G))) and  
 DELCL(((ejec oui))) and  
 ADDCL(((AugmNv -))) and  
 ADDCL(((Nv MG))) and  
 ADDCL(((ejec non))) and  
 dimRao and  
 /

Voir remarque 1.

100. Nv-apres(MG) if  
 AugmNv(+) and  
 Nv(G) and  
 Emax(norm) and  
 DELCL(((Nv G))) and  
 DELCL(((AugmNv +))) and  
 ADDCL(((Nv MG))) and  
 ADDCL(((AugmNv -))) and  
 dimRao and  
 /

Voir remarque 1.

101. Nv-apres(P) if  
 AugmNv(-) and  
 Nv(MG) and  
 Emax(norm) and  
 DELCL(((Nv MG))) and  
 ADDCL(((Nv P)))

Voir remarque 1.



102. Nv-apres(0) if  
 AugmNv(-) and  
 Nv(MP) and  
 Emax(grand) and  
 DELCL(((Nv MP))) and  
 DELCL(((AugmNv -))) and  
 DELCL(((repos))) and  
 ADDCL(((Nv 0))) and  
 ADDCL(((AugmNv 0))) and  
 ADDCL(((repos debut)))

Voir remarque 1.

103. Nv-apres(MP) if  
 Nv(0) and  
 repos(fin) and  
 Emax(grand) and  
 DELCL(((repos fin))) and  
 DELCL(((Nv 0))) and  
 DELCL(((AugmNv 0))) and  
 ADDCL(((repos))) and  
 ADDCL(((Nv MP))) and  
 ADDCL(((AugmNv +)))

Voir remarque 1.

104. Nv-apres(MG) if  
 AugmNv(+) and  
 Nv(MP) and  
 Emax(grand) and  
 DELCL(((Nv MP))) and  
 ADDCL(((Nv MG)))

Voir remarque 1.

105. Nv-apres(G) if  
 AugmNv(+) and  
 Nv(MG) and  
 Emax(grand) and  
 DELCL(((Nv MG))) and  
 ADDCL(((Nv G)))

Voir remarque 1.

106. Nv-apres(TG) if  
 AugmNv(+) and  
 Nv(G) and  
 Emax(grand) and  
 DELCL(((Nv G))) and  
 ADDCL(((Nv TG))) and  
 augmRao

Voir remarque 1.

107. Nv-apres(G) if  
 AugmNv(+) and  
 Nv(TG) and  
 Emax(grand) and  
 DELCL(((AugmNv +))) and  
 DELCL(((Nv TG))) and  
 DELCL(((ejec oui))) and  
 ADDCL(((AugmNv -))) and  
 ADDCL(((Nv G))) and  
 ADDCL(((ejec non))) and  
 dimRao and  
 /

Voir remarque 1.

108. Nv-apres(G) if  
 AugmNv(+) and  
 Nv(TG) and  
 Emax(grand) and  
 DELCL(((Nv TG))) and  
 DELCL(((AugmNv +))) and  
 ADDCL(((Nv G))) and  
 ADDCL(((AugmNv -))) and  
 dimRao and  
 /

Voir remarque 1.



109. Nv-apres(MP) if  
    AugmNv(-) and  
    Nv(G) and  
    Emax(grand) and  
    DELCL(((Nv G))) and  
    ADDCL(((Nv MP)))

Voir remarque 1.

110. Nv-apres(O) if  
    AugmNv(-) and  
    Nv(TP) and  
    Emax(petit) and  
    DELCL(((Nv TP))) and  
    DELCL(((AugmNv -))) and  
    DELCL(((repos))) and  
    ADDCL(((Nv O))) and  
    ADDCL(((AugmNv O))) and  
    ADDCL(((repos debut)))

Voir remarque 1.

111. Nv-apres(TP) if  
    Nv(O) and  
    repos(fin) and  
    Emax(petit) and  
    DELCL(((repos fin))) and  
    DELCL(((Nv O))) and  
    DELCL(((AugmNv O))) and  
    ADDCL(((repos))) and  
    ADDCL(((Nv TP))) and  
    ADDCL(((AugmNv +)))

Voir remarque 1.

112. Nv-apres(P) if  
    AugmNv(+) and  
    Nv(TP) and  
    Emax(petit) and  
    DELCL(((Nv TP))) and  
    ADDCL(((Nv P)))

Voir remarque 1.

113. Nv-apres(MP) if  
    AugmNv(+) and  
    Nv(P) and  
    Emax(petit) and  
    DELCL(((Nv P))) and  
    ADDCL(((Nv MP)))

Voir remarque 1.

114. Nv-apres(MG) if  
    AugmNv(+) and  
    Nv(MP) and  
    Emax(peti) and  
    DELCL(((Nv MP))) and  
    ADDCL(((Nv MG))) and  
    augmRao and  
    /

Voir remarque 1.

115. Nv-apres(MP) if  
    AugmNv(+) and  
    Nv(MG) and  
    Emax(petit) and  
    DELCL(((Nv MG))) and  
    DELCL(((AugmNv +))) and  
    DELCL(((ejeç oui))) and  
    ADDCL(((Nv MP))) and  
    ADDCL(((AugmNv -))) and  
    ADDCL(((ejeç non))) and  
    dimRao and  
    /

Voir remarque 1.



116. Nv-apres(MP) if  
 AugmNv(+) and  
 Nv(MG) and  
 Emax(petit)) and  
 DELCL(((Nv MG))) and  
 DELCL(((AugmNv +))) and  
 ADDCL(((Nv MP))) and  
 ADDCL(((AugmNv -))) and  
 dimRao and  
 /

Voir remarque 1.

117. Nv-apres(TP) if  
 AugmNv(-) and  
 Nv(MP) and  
 Emax(petit) and  
 DELCL(((Nv MP))) and  
 ADDCL(((Nv TP)))

Voir remarque 1.

#### remarque 1

#### Description générale de la propagation dans le temps du nombre d'élastiques autour du coeur

- \* Lorsque le ventricule gauche, qui est en début de phase de repos (repos(debut)), passe en fin de phase de repos (repos(fin)), il n'y a aucun effet sur le nombre d'élastiques autour du coeur qui conserve une valeur minimale (Nv(0)) et le coeur gauche est au repos (AugmNv(0)).
- \* Lorsque le nombre d'élastiques autour du coeur augmente le coeur gauche passe en activité croissante (AugmNv(+)) et le ventricule gauche n'est plus en phase de repos (repos()).
- \* Lorsque le nombre d'élastiques autour du coeur atteint sa valeur maximale, la résistance passante de la valve aortique est augmentée (augmRao). Cette augmentation constitue une hypothèse de modélisation. Elle permet d'observer des résultats d'exécution compatibles au réel perçu.
- \* Lorsque le nombre d'élastiques autour du coeur perd sa valeur maximale, le coeur gauche passe en activité décrois-

sante (AugmNv(-)) et la résistance passante de la valve aortique est rétablie à sa valeur initiale (dimRao). Cette diminution constitue une hypothèse de modélisation. Elle permet de rétablir la résistance passante de la valve aortique à sa valeur initiale.

- \* Lorsque le coeur gauche est en activité décroissante et que le nombre d'élastiques autour du coeur prend une valeur minimale, le ventricule gauche commence à se remplir et le coeur gauche est au repos (AugmNv(0)) (repos(debut)).

```
118. dt(G) if
      Ryhme(norm) and
      repos(debut) and
      /
```

L'intervalle de temps à considérer lors de la propagation du temps est G

Si le rythme cardiaque est normal (Ryhme(norm))

ET

Si le ventricule gauche est en début de remplissage (repos(debut)).

```
119. dt(TP) if
      Ryhme(norm) and
      AugmNv(+) and
      /
```

L'intervalle de temps à considérer lors de la propagation du temps est TP

Si le rythme cardiaque est normal (Ryhme(norm))

ET

Si le nombre d'élastiques autour du coeur augmente (AugmNv(+)).



120. dt(TP) if  
    Rythme(court) and  
    not(repos debut) and  
    /  
    /

L'intervalle de temps à considérer lors de la propagation du temps est TP

Si le rythme cardiaque est court (Rythme(court))

ET

Si le ventricule gauche n'est pas en début de phase de repos (not repos(debut)).

121. dt(TP) if  
    Rythme(norm) and  
    Emax(norm) and  
    AugmNv(-) and  
    Nv(P) and  
    /  
    /

L'intervalle de temps à considérer lors de la propagation du temps est TP

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est normale (Emax(norm))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est P (Nv(P)).

122. dt(TP) if

Rythme(norm) and  
Emax(grand) and  
AugmNv(-) and  
Nv(MP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est TP

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est grande par rapport à la normale (Emax(grand))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MP (Nv(MP)).

123. dt(TP) if

Rythme(norm) and  
Emax(petit) and  
AugmNv(-) and  
Nv(TP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est TP

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est petite par rapport à la normale (Emax(petit))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est TP (Nv(TP)).



124. dt(P) if  
    Rythme(norm) and  
    repos(fin) and  
    /

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est normal (Rythme(norm))

ET

Si le ventricule gauche est en fin de phase de repos (repos(fin)).

125. dt(P) if  
    Rythme(long) and  
    AugmNv(+) and  
    /

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

ET

Si le nombre d'élastiques autour du coeur augmente (AugmNv(+)).

126. dt(P) if

Rythme(norm) and  
Emax(norm) and  
AugmNv(-) and  
Nv(MG) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est normale (Emax(norm))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MG (Nv(MG)).

127. dt(P) if

Rythme(norm) and  
Emax(grand) and  
AugmNv(-) and  
Nv(G) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est grande par rapport à la normale (Emax(grand))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est G (Nv(G))



128. dt(P) if

Rythme(norm) and  
Emax(petit) and  
AugmNv(-) and  
Nv(MP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est normal (Rythme(norm))

Si l'élastance du ventricule est petite par rapport à la normale (Emax(petit))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MP (Nv(MP)).

129. dt(P) if

Rythme(long) and  
Emax(norm) and  
AugmNv(-) and  
Nv(P) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

Si l'élastance du ventricule est normale (Emax(norm))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est P (Nv(P))

130. dt(P) if

Rythme(long) and  
Emax(grand) and  
AugmNv(-) and  
Nv(MP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

Si l'élastance du ventricule est grande par rapport à la normale (Emax(grand))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MP (Nv(MP))

131. dt(P) if

Rythme(long) and  
Emax(petit) and  
AugmNv(-) and  
Nv(TP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est P

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

Si l'élastance du ventricule est petite par rapport à la normale (Emax(petit))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est TP (Nv(TP))



132. dt(MP) if

Rythme(long) and  
repos(fin) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est MP

Si le rythme cardiaque est long (Rythme(long))

ET

Si le ventricule gauche est en fin de phase de repos (repos(fin)).

133. dt(MP) if

Rythme(long) and  
Emax(norm) and  
AugmNv(-) and  
Nv(MG) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est MP

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

Si l'élastance du ventricule est normale (Emax(norm))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MP (Nv(MP))

134. dt(MP) if

Rythme(long) and  
Emax(grand) and  
AugmNv(-) and  
Nv(G) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est MP

Si le rythme cardiaque est long par rapport à la normale (Rythme(long))

Si l'élastance du ventricule est grande par rapport à la normale (Emax(grand))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est G (Nv(G))

135. dt(MP) if

Rythme(long) and  
Emax(petit) and  
AugmNv(-) and  
Nv(MP) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est MP

Si le rythme cardiaque est court par rapport à la normale (Rythme(court))

Si l'élastance du ventricule est petite par rapport à la normale (Emax(petit))

Si le nombre d'élastiques autour du coeur diminue (AugmNv(-))

ET

Si le nombre d'élastiques autour du coeur est MP (Nv(MP)).



136. dt(MG) if

Rythme(court) and  
repos(debut) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est MG

Si le rythme cardiaque est court (Rythme(court))

ET

Si le ventricule gauche est en début de phase de repos (repos(debut)).

137. dt(TG) if

Rythme(long) and  
repos(debut) and  
/

L'intervalle de temps à considérer lors de la propagation du temps est TG

Si le rythme cardiaque est long (Rythme(long))

ET

Si le ventricule gauche est en début de phase de repos (repos(debut)).

138. augmRao if

DELCL(((Raorte-pas O))) and  
ADDCL(((Raorte-pas P))) and  
ADDCL(((artifice Rao))) and  
/

Voir remarque 2.

139. augmRao if

DELCL(((Raorte-pas TP))) and  
ADDCL(((Raorte-pas P))) and  
/

Voir remarque 2.

140.augmRao if

DELCL(((Raorte-pas P))) and  
ADDCL(((Raorte-pas MP))) and  
/

Voir remarque 2.

141.augmRao if

DELCL(((Raorte-pas MP))) and  
ADDCL(((Raorte-pas MG))) and  
/

Voir remarque 2.

142.augmRao if

DELCL(((Raorte-pas MG))) and  
ADDCL(((Raorte-pas G))) and  
/

Voir remarque 2.

143.augmRao if

DELCL(((Raorte-pas G))) and  
ADDCL(((Raorte-pas TG))) and  
ADDCL(((artifice RaoG))) and  
/

Voir remarque 2.

144.augmRao if

/

Voir remarque 2.

### remarque 2

Les règles 138 à 144 ont pour effet d'augmenter la valeur de la résistance passante de la valve aortique lors de la dernière éjection ventriculaire. Cette augmentation constitue une hypothèse de modélisation. Elle permet l'observation de résultats d'exécution compatibles au réel perçu.



Description générale de l'augmentation de la résistance pas-  
sante de la valve aortique

- \* En général, lors de l'augmentation de la résistance passante de la valve aortique, la résistance prend la valeur qualitative suivante dans l'échelle de valeurs définie (0 TP P MP MG G TG). Cependant, dans le cas où la valeur de la résistance est 0, elle passe directement à P et dans le cas où cette valeur est TG, elle n'est pas modifiée.
- \* Il est nécessaire d'introduire un artifice de programmation dans la première d'augmentation dans la mesure où la résistance peut prendre la valeur P si à l'origine elle vaut 0 ou TP. Lorsqu'il faudra rétablir la valeur de la résistance, il sera nécessaire de savoir si elle était égale à 0 ou TP. C'est pourquoi, dans le cas où la valeur initiale de la résistance est 0 un fait le renseignant est introduit dans la base de connaissances ( ((ADDCL augmRao Rao)) ). De même, il est nécessaire d'introduire un artifice de programmation dans l'avant-dernière augmentation dans la mesure où la résistance peut prendre la valeur TG si à l'origine elle vaut G ou TG. Lorsqu'il faudra rétablir la valeur de la résistance, il sera nécessaire de savoir si elle était égale à G ou TG. C'est pourquoi, dans le cas où la valeur initiale de la résistance est G un fait le renseignant est introduit dans la base de connaissances (((ADDCL augmRao RaoG))).
- \* Dans toutes les règles d'augmentation de la résistance passante de la valve aortique, il est nécessaire d'introduire le prédicat "cut" afin d'empêcher toute nouvelle tentative qui conduirait à une série d'augmentations jusqu'à ce que la résistance atteigne la valeur TG.

```
145.dimRao if
    DELCL(((artifice Rao))) and
    DELCL(((Raorte-pas P))) and
    ADDCL(((Raorte-pas 0))) and
    /
```

Voir remarque 3.

146.dimRao if  
DELCL(((Raorte-pas P))) and  
ADDCL(((Raorte-pas TP))) and  
/

Voir remarque 3.

147.dimRao if  
DELCL(((Raorte-pas MP))) and  
ADDCL(((Raorte-pas P))) and  
/

Voir remarque 3.

148.dimRao if  
DELCL(((Raorte-pas MG))) and  
ADDCL(((Raorte-pas MP))) and  
/

Voir remarque 3.

149.dimRao if  
DELCL(((Raorte-pas G))) and  
ADDCL(((Raorte-pas MG))) and  
/

Voir remarque 3.

150.dimRao if  
DELCL(((artifice RaoG))) and  
DELCL(((Raorte-pas TG))) and  
ADDCL(((Raorte-pas G))) and  
/

Voir remarque 3.

151.dimRao if  
/

Voir remarque 3.



### remarque 3

Les règles 145 à 151 ont pour effet de diminuer la valeur de la résistance passante de la valve aortique lorsque le coeur gauche passe en phase de relaxation. Cette diminution constitue une hypothèse de modélisation. Elle permet de rétablir la valeur initiale de la résistance passante de la valve aortique qui a été augmentée lors de l'éjection.

### Description générale de la diminution de la résistance passante de la valve aortique

- \* En général, lors de la diminution de la résistance passante de la valve aortique, la résistance prend la valeur qualitative précédente dans l'échelle de valeurs définie (0 TP P MP MG G TG). Cependant, dans le cas où la valeur de la résistance est P il convient de s'assurer que cette valeur provient de l'augmentation antérieure d'une résistance dont la valeur était 0 de manière à la rétablir. De même, dans le cas où la valeur de la résistance est TG il convient de s'assurer que cette valeur provient de l'augmentation antérieure d'une résistance dont la valeur était G de manière à la rétablir.
- \* Dans toutes les règles de diminution de la résistance passante de la valve aortique il est nécessaire d'introduire le prédicat "cut" afin d'empêcher toute nouvelle tentative qui conduirait à une série de diminutions jusqu'à ce que la résistance atteigne la valeur TP.

```
152.vol-vent-resultant if
    Volv(X) and
    var-Volv(Y) and
    additv(X Y Z)
    PP(Z)
```

Règle permettant à l'utilisateur de demander le volume de sang dans le ventricule ajusté à la variation de ce même volume dans la phase.

153.vol-aorte-resultant if  
Volao(X) and  
var-Volao(Y) and  
additao(X Y Z) and  
PP(Z)

Règle permettant à l'utilisateur de demander le volume de sang dans l'aorte ajusté à la variation de ce même volume dans la phase.

154.Pa-cont-vent(MP) if  
Pa(0) and  
/

Règle de transition de valeur du contexte de l'aorte dans le contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est égale à MP si elle a la valeur 0 dans son contexte.

155.Pa-cont-vent(MP) if  
Pa(TP) and  
/

Règle de transition de valeur du contexte de l'aorte dans le contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est égale à MP si elle a la valeur TP dans son contexte.

156.Pa-cont-vent(MG) if  
Pa(P) and  
/

Règle de transition de valeur du contexte de l'aorte dans le contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est égale à MG si elle a la valeur P dans son contexte.



157.Pa-cont-vent(MG) if  
Pa(MP) and  
/

Règle de transition de valeur du contexte de l'aorte dans le  
contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est  
égale à MG si elle a la valeur MP dans son contexte.

158.Pa-cont-vent(MG) if  
Pa(MG) and  
/

Règle de transition de valeur du contexte de l'aorte dans le  
contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est  
égale à MG si elle a la valeur MG dans son contexte.

159.Pa-cont-vent(G) if  
Pa(G) and  
/

Règle de transition de valeur du contexte de l'aorte dans le  
contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est  
égale à G si elle a la valeur G dans son contexte.

160.Pa-cont-vent(TG) if  
Pa(TG) and  
/

Règle de transition de valeur du contexte de l'aorte dans le  
contexte du ventricule.

La pression de l'aorte dans le contexte du ventricule est  
égale à TG si elle a la valeur TG dans son contexte.

161.Pv-cont-aorte(O) if  
Pv(O) and  
/

Règle de transition de valeur du contexte du ventricule dans  
le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à 0 si elle a la valeur 0 dans son contexte.

162.Pv-cont-aorte(0) if  
Pv(TP) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à 0 si elle a la valeur TP dans son contexte.

163.Pv-cont-aorte(0) if  
Pv(P) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à 0 si elle a la valeur P dans son contexte.

164.Pv-cont-aorte(TP) if  
Pv(MP) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à TP si elle a la valeur MP dans son contexte.

165.Pv-cont-aorte(MG) if  
Pv(MG) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à MG si elle a la valeur MG dans son contexte.



166.Pv-cont-aorte(G) if  
Pv(G) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à G si elle a la valeur G dans son contexte.

167.Pv-cont-aorte(TG) if  
Pv(TG) and  
/

Règle de transition de valeur du contexte du ventricule dans le contexte de l'aorte.

La pression du ventricule dans le contexte de l'aorte est égale à TG si elle a la valeur TG dans son contexte.

168.pos(X) if  
X EQ TP

Voir remarque 4.

169.pos(X) if  
X EQ P

Voir remarque 4.

170.pos(X) if  
X EQ MP

Voir remarque 4.

171.pos(X) if  
X EQ MG

Voir remarque 4.

172.pos(X) if  
X EQ G

Voir remarque 4.

```
173.pos(X)  if
           X EQ TG
```

Voir remarque 4.

remarque 4

Une variable est positive si elle prend une valeur dans l'échelle de valeurs définie (0 TP P MP MG G TG).

```
174.plus-petit(X Y)  if
                     diffn(X Y Z) and
                     pos(Z)
```

X est plus petit que Y si  $Y - X$  est positif.

```
175.plus-grand-eg(X Y)  if
                       diffn(X Y Z) and
                       pos(Z)
```

X est plus grand ou égal à Y si  $X - Y$  est positif.

```
176.plus-grand-eg(X Y)  if
                       diffn(X Y) and
                       Z EQ 0
```

X est plus grand ou égal à Y si  $X - Y$  est nul.

```
177.help  if
          PP(MENU :) and
          PP(is(help-ventricule))
          PP(is(help-aorte))
          PP(is(help-debit))
          PP(is(help-phase))
          PP(is(help-parametres))
```

Règle permettant à l'utilisateur de connaître les différents domaines sur lesquels il peut interroger le système.



178.help-ventricule if

```
PP(LISTE DES COMMANDES POSSIBLES : ) and
PP(is(vol-vent)) and
PP(is(pres-vent)) and
PP(is(debit-valve-mitrale)) and
PP(is(debit-valve-aortique)) and
PP(is(variation-vol-vent)) and
PP(is(raidur-vent)) and
PP(is(remplissage)) and
PP(is(ejection))
```

Règle permettant à l'utilisateur de connaître les questions qu'il peut poser au système à propos du ventricule.

179.help-aorte if

```
PP(LISTE DES COMMANDES POSSIBLES : ) and
PP(is(vol-aorte)) and
PP(is(pres-aorte)) and
PP(is(debit-aortique)) and
PP(is(debit-reseau-periph)) and
PP(is(variation-vol-aorte)) and
PP(is(ejection))
```

Règle permettant à l'utilisateur de connaître les questions qu'il peut poser au système à propos de l'aorte.

180.help-debit if

```
PP(LISTE DES COMMANDES POSSIBLES : ) and
PP(is(debit-valve-mitrale)) and
PP(is(debit-valve-aortique)) and
PP(is(debit-reseau-periph))
```

Règle permettant à l'utilisateur de connaître les questions qu'il peut poser au système à propos des débits sanguins.

181.help-phase if

```
PP(LISTE DES COMMANDES POSSIBLES :) and
PP(is(phase-coeur)) and
PP(is(mitracle-ouverte)) and
PP(is(mitracle-fermee)) and
PP(is(aortique-ouverte)) and
PP(is(aortique-fermee)) and
PP(is(remplissage)) and
PP(is(ejection)) and
PP(is(activite-croiss)) and
PP(is(activite-decroiss))
```

Règle permettant à l'utilisateur de connaître les questions qu'il peut poser au système à propos des phases cardiaques.

182.help-parametres if

```
PP(LISTE DES COMMANDES POSSIBLES : ) and
PP(which(X:Pres(X))) and
PP(which(X:Rmitrale-pas(X))) and
PP(which(X:Rmitrale-bloc(X))) and
PP(which(X:Raortique-pas(X))) and
PP(which(X:Raortique-bloc(X))) and
PP(which(X:Emax(X))) and
PP(which(X:Rythme(X))) and
PP(which(X:Rper(X))) and
PP(which(X:raid-a(X))) and
PP(which(X:Nv(X)))
```

Règle permettant à l'utilisateur de connaître les paramètres du système dont il peut obtenir la valeur.



## ANNEXE 4

### PROGRAMME LISP

#### ENREGISTREMENT DES VALEURS DES PARAMETRES

1. (SETQ VOLV 'O)
2. (SETQ NV 'O)
3. (SETQ VOLAO 'MG)
4. (SETQ RAID 'MP)
5. (SETQ EJEC NIL)
6. (SETQ PRES 'P)
7. (SETQ RMITRALEPAS 'O)
8. (SETQ RMITRALEBLOC 'TG)
9. (SETQ RPER 'MG)
10. (SETQ RAORTIQUEPAS 'O)
11. (SETQ RAORTIQUEBLOC 'TG)
12. (SETQ PSYS 'O)
13. (SETQ RCAR 'TP)
14. (SETQ PV NIL)
15. (SETQ PA NIL)
16. (SETQ AUGMNV 'O)
17. (SETQ RYTHMEC 'NORM)
18. (SETQ EMAX 'NORM)
19. (SETQ REPOS 'DEBUT)

#### ENREGISTREMENT DES OPERATIONS QUALITATIVES

20. (SETQ DIVAOPE '  
((O ((O P) (TP TP) (P TP) (MP O) (MG O) (G O) (TG O)))  
(TP ((O MG) (TP MP) (P MP) (MP P) (MG P) (G TP) (TG O)))  
(P ((O G) (TP G) (P MG) (MP MG) (MG MP) (G P) (TG TP)))  
(MP ((O TG) (TP G) (P G) (MP MG) (MG MP) (G MP) (TG P)))  
(MG ((O TG) (TP TG) (P G) (MP G) (MG MG) (G MG) (TG MP)))  
(G ((O TG) (TP TG) (P TG) (MP G) (MG G) (G MG) (TG MP)))  
(TG ((O TG) (TP TG) (P TG) (MP TG) (MG G) (G G) (TG MG))))))

21. (SETQ DIVVPE ')

```
((O ((O TP) (TP O) (P O) (MP O) (MG O) (G O) (TG O)))
(TP ((O MP) (TP MP) (P P) (MP P) (MG TP) (G TP) (TG O)))
(P ((O G) (TP G) (P MG) (MP MP) (MG P) (G TP) (TG TP)))
(MP ((O G) (TP G) (P MG) (MP MG) (MG MP) (G P) (TG TP)))
(MG ((O TG) (TP G) (P G) (MP MG) (MG MG) (G MP) (TG P)))
(G ((O TG) (TP TG) (P G) (MP G) (MG MG) (G MP) (TG MP)))
(TG ((O TG) (TP TG) (P TG) (MP G) (MG G) (G MG) (TG MP))))
```

22. (SETQ DIFFN ')

```
((O ((O O) (TP -TP) (P -P) (MP -MP) (MG -MG) (G -G)
(TG -TG)))
(TP ((O TP) (TP O) (P -TP) (MP -P) (MG -MP) (G -MG)
(TG -G)))
(P ((O P) (TP TP) (P O) (MP -TP) (MG -P) (G -MP)
(TG -MG)))
(MP ((O MP) (TP P) (P TP) (MP O) (MG -TP) (G -P)
(TG -MP)))
(MG ((O MG) (TP MP) (P P) (MP TP) (MG O) (G -TP)
(TG -P)))
(G ((O G) (TP MG) (P MP) (MP P) (MG TP) (G O) (TG -TP)))
(TG ((O TG) (TP G) (P MG) (MP MP) (MG P) (G TP) (TG O))))
```

23. (SETQ DIVPR ')

```
((O ((O O) (TP O) (P O) (MP O) (MG O) (G O) (TG O)))
(TP ((O TP) (TP TP) (P TP) (MP TP) (MG TP) (G TP) (TG O)))
(P ((O P) (TP P) (P TP) (MP TP) (MG TP) (G TP) (TG O)))
(MP ((O MP) (TP MP) (P P) (MP TP) (MG TP) (G TP) (TG O)))
(MG ((O MG) (TP MG) (P MP) (MP P) (MG TP) (G TP) (TG O)))
(G ((O G) (TP G) (P MG) (MP MP) (MG P) (G TP) (TG O)))
(TG ((O TG) (TP TG) (P G) (MP MG) (MG MP) (G P) (TG O)))
(-TP ((O -TP) (TP -TP) (P -TP) (MP -TP) (MG -TP) (G -TP)
(TG O)))
(-P ((O -P) (TP -P) (P -TP) (MP -TP) (MG -TP) (G -TP)
(TG O)))
(-MP ((O -MP) (TP -MP) (P -P) (MP -TP) (MG -TP) (G -TP)
(TG O)))
(-MG ((O -MG) (TP -MG) (P -MP) (MP -P) (MG -TP) (G -TP)
(TG O)))
(-G ((O -G) (TP -G) (P -MG) (MP -MP) (MG -P) (G -TP)
(TG O)))
(-TG ((O -TG) (TP -TG) (P -G) (MP -MG) (MG -MP) (G -P)
(TG O))))
```



24. (SETQ ADDIT ')

```
((O ((O O) (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(TP ((O TP) (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(P ((O P) (TP P) (P MP) (MP MG) (MG G) (G TG) (TG TG)))
(MP ((O MP) (TP MP) (P MG) (MP G) (MG G) (G TG) (TG TG)))
(MG ((O MG) (TP MG) (P G) (MP G) (MG TG) (G TG) (TG TG)))
(G ((O G) (TP G) (P TG) (MP TG) (MG TG) (G TG) (TG TG)))
(TG ((O TG) (TP TG) (P TG) (MP TG) (MG TG) (G TG) (TG TG))))
```

25. (SETQ MULTVQE ')

```
((O ((O O) (TP TP) (P TP) (MP P) (MG P) (G MP) (TG MP)))
(TP ((O O) (TP TP) (P TP) (MP P) (MG MP) (G MG) (TG MG)))
(P ((O TP) (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(MP ((O TP) (TP P) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(MG ((O TP) (TP P) (P MP) (MP MP) (MG MG) (G G) (TG TG)))
(G ((O P) (TP P) (P MP) (MP MP) (MG MG) (G TG) (TG TG)))
(TG ((O P) (TP MP) (P MP) (MP MG) (MG G) (G TG) (TG TG))))
```

26. (SETQ MULTAOQE ')

```
((O ((O O) (TP O) (P TP) (MP TP) (MG TP) (G P) (TG P)))
(TP ((O O) (TP TP) (P TP) (MP P) (MG P) (G P) (TG MP)))
(P ((O O) (TP TP) (P P) (MP P) (MG MP) (G MP) (TG MG)))
(MP ((O TP) (TP P) (P P) (MP MP) (MG MG) (G MG) (TG G)))
(MG ((O TP) (TP P) (P MP) (MP MG) (MG MG) (G G) (TG G)))
(G ((O P) (TP MP) (P MG) (MP G) (MG G) (G TG) (TG TG)))
(TG ((O P) (TP MP) (P MG) (MP G) (MG TG) (G TG) (TG TG))))
```

27. (SETQ MULTAOAPRES ')

```
((-TG ((TP -TG) (P -TG) (MP -TG) (MG -TG) (G -TG)
(TG -TG)))
(-G ((TP -G) (P -G) (MP -G) (MG -TG) (G -TG) (TG -TG)))
(-MG ((TP -MG) (P -MG) (MP -MG) (MG -G) (G -G) (TG -G)))
(-MP ((TP -MP) (P -MP) (MP -MP) (MG -MG) (G -MG)
(TG -G)))
(-P ((TP -P) (P -P) (MP -P) (MG -MP) (G -MP) (TG -MP)))
(-TP ((TP -TP) (P -P) (MP -P) (MG -P) (G -P) (TG -MP)))
(O ((TP O) (P O) (MP O) (MG O) (G O) (TG O)))
(TP ((TP TP) (P P) (MP P) (MG P) (G P) (TG MP)))
(P ((TP P) (P P) (MP P) (MG MP) (G MP) (TG MP)))
(MP ((TP MP) (P MP) (MP MP) (MG MG) (G MG) (TG G)))
(MG ((TP MG) (P MG) (MP MG) (MG G) (G G) (TG G)))
(G ((TP G) (P G) (MP G) (MG TG) (G TG) (TG TG)))
(TG ((TP TG) (P TG) (MP TG) (MG TG) (G TG) (TG TG))))
```



28. (SETQ MULTVAPRES ')

```
((-TG ((TP -TG) (P -TG) (MP -TG) (MG -TG) (G -TG)
      (TG -TG)))
  (-G ((TP -G) (P -G) (MP -G) (MG -G) (G -T) (TG -G)))
  (-MG ((TP -MG) (P -G) (MP -G) (MG -G) (G -G) (TG -TG)))
  (-MP ((TP -MP) (P -MG) (MP -MG) (MG -MG) (G -G) (TG -G)))
  (-P ((TP -P) (P -MP) (MP -MG) (MG -MG) (G -G) (TG -G)))
  (-TP ((TP -TP) (P -P) (MP -MP) (MG -MG) (G -G) (TG -G)))
  (O ((TP O) (P O) (MP O) (MG O) (G O) (TG O)))
  (TP ((TP TP) (P P) (MP MP) (MG MG) (G G) (TG G)))
  (P ((TP P) (P MP) (MP MG) (MG MG) (G G) (TG G)))
  (MP ((TP MP) (P MG) (MP MG) (MG MG) (G G) (TG G)))
  (MG ((TP MG) (P G) (MP G) (MG G) (G G) (TG TG)))
  (G ((TP G) (P G) (MP G) (MG G) (G G) (TG TG)))
  (TG ((TP TG) (P TG) (MP TG) (MG TG) (G TG) (TG TG))))
```

29. (SETQ ADDAOAPRES ')

```
((O ((-TG O) (-G O) (-MG O) (-MP O) (-P O) (-TP O) (O O)
      (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
  (TP ((-TG O) (-G O) (-MG O) (-MP O) (-P O) (-TP TP)
      (O TP) (TP P) (P MP) (MP MG) (MG MG) (G G) (TG TG)))
  (P ((-TG O) (-G O) (-MG O) (-MP O) (-P TP) (-TP P) (O P)
      (TP MP) (P MG) (MP G) (MG G) (G TG) (TG TG)))
  (MP ((-TG O) (-G O) (-MG O) (-MP TP) (-P P) (-TP MP)
      (O MP) (TP MG) (P G) (MP G) (MG G) (G TG) (TG TG)))
  (MG ((-TG O) (-G O) (-MG TP) (-MP P) (-P MP) (-TP MG)
      (O MG) (TP MG) (P G) (MP G) (MG G) (G TG) (TG TG)))
  (G ((-TG O) (-G TP) (-MG P) (-MP MP) (-P MG) (-TP G)
      (O G) (TP G) (P TG) (MP TG) (MG TG) (G TG) (TG TG)))
  (TG ((-TG TP) (-G P) (-MG MP) (-MP MG) (-P G) (-TP TG)
      (O TG) (TP TG) (P TG) (MP TG) (MG TG) (G TG)
      (TG TG))))
```



30. (SETQ ADDVAPRES ')

```
((O ((-TG O) (-G O) (-MG O) (-MP O) (-P O) (-TP O) (O O)
      (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(TP ((-TG O) (-G O) (-MG O) (-MP O) (-P O) (-TP O)
      (O TP) (TP TP) (P P) (MP MP) (MG MG) (G G) (TG TG)))
(P ((-TG O) (-G O) (-MG O) (-MP O) (-P TP) (-TP TP)
      (O P) (TP P) (P MP) (MP MG) (MG G) (G G) (TG TG)))
(MP ((-TG O) (-G O) (-MG O) (-MP TP) (-P P) (-TP P)
      (O MP) (TP MP) (P MG) (MP G) (MG G) (G TG) (TG TG)))
(MG ((-TG O) (-G O) (-MG TP) (-MP P) (-P P) (-TP MP)
      (O MG) (TP MG) (P G) (MP G) (MG G) (G TG) (TG TG)))
(G ((-TG O) (-G TP) (-MG P) (-MP P) (-P MP) (-TP MG)
      (O G) (TP G) (P G) (MP TG) (MG TG) (G TG) (TG TG)))
(TG ((-TG TP) (-G P) (-MG MP) (-MP MP) (-P MG) (-TP G)
      (O TG) (TP TG) (P TG) (MP TG) (MG TG) (G TG)
      (TG TG))))
```

#### ENREGISTREMENT DES CORRESPONDANCES DE VALEURS ENTRE CONTEXTES

31. (SETQ PVCONTAO ')

```
((O O) (TP O) (P O) (MP TP) (MG MG) (G G) (TG TG) )
```

32. (SETQ PACONTV ')

```
((O MP) (TP MP) (P MG) (MP MG) (MG MG) (G G) (TG TG) )
```

#### ENREGISTREMENT DE L'ORDRE DE GRANDEUR DES VALEURS QUALITATIVES

33. (SETQ PPETIT ')

```
((O (O TP P MP MG G TG))
(TP (TP P MP MG G TG))
(P (P MP MG G TG))
(MP (MP MG G TG))
(MG (MG G TG))
(G (G TG))
(TG (TG) ) )
```

# FONCTIONS RECHERCHANT LA SOLUTION A UN CALCUL QUALITATIF

34. (DEF 'MULTV '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X MULTVQE)))))) )
35. (DEF 'MULTAO '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X MULTAOQE)))))) )
36. (DEF 'DIV '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X DIVPR)))))) )
37. (DEF 'DIFF '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X DIFFN)))))) )
38. (DEF 'DIV '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X DIVPR)))))) )
39. (DEF 'ADD '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X ADDIT)))))) )
40. (DEF 'DIVV '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X DIVVPE)))))) )
41. (DEF 'DIVAO '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X DIVAOPE)))))) )
42. (DEF 'ADDVAP '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X ADDVAPRES)))))) )
43. (DEF 'MULTVAP '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X MULTVAPRES)))))) )
44. (DEF 'ADDAOAP '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X ADDAOAPRES)))))) )
45. (DEF 'MULTAOAP '(LAMBDA (X Y)  
          (CADR (ASSOC Y (CADR (ASSOC X MULTAOAPRES)))))) )



FONCTION CALCULANT LA PRESSION DU VENTRICULE

```
46. (DEF 'PRESV ' (LAMBDA (VOLV NV)
      (COND ((NULL VOLV) PV)
            (T (MULTV VOLV NV) ))))
```

FONCTION CALCULANT LA PRESSION DE L'AORTE

```
47. (DEF 'PRESAO ' (LAMBDA (VOLAO RAID)
      (COND ((EQUAL EJEC T) (CONVERSIONPV())) )
      ((NULL VOLAO) PA)
      (T (MULTAO VOLAO RAID) ))))
```

FONCTION TESTANT SI UNE VALEUR QUALITATIVE EST PLUS PETITE OU NON QU'UNE AUTRE

```
48. (DEF 'PLUSPETIT ' (LAMBDA (X Y)
      (COND ((MEMB Y (CADR (ASSOC X PPETIT)) ) T)
            (T NIL) ) ))
```

FONCTION EFFECTUANT LA CONVERSION DE LA VALEUR DE LA PRESSION DU VENTRICULE DANS LE CONTEXTE DE L'AORTE

```
49. (DEF 'CONVERSIONPV ' (LAMBDA ()
      (CADR ( ASSOC (PRESV VOLV NV) PVCONTAO))))
```

FONCTION EFFECTUANT LA CONVERSION DE LA VALEUR DE LA PRESSION DE L'AORTE DANS LE CONTEXTE DU VENTRICULE

```
50. (DEF 'CONVERSIONPA ' (LAMBDA ()
      (CADR (ASSOC (PRESAO VOLAO RAID) PACONTV))))
```

FONCTION DETERMINANT SI LA VALVE AORTIQUE EST OUVERTE OU PAS

```
51. (DEF 'AORTIQUEOUV ' (LAMBDA ()
      (PLUSPETIT (CONVERSIONPA()) (PRESV VOLV NV))))
```

FONCTION DETERMINANT SI LA VALVE MITRALE EST OUVERTE OU PAS

```
52. (DEF 'MITRALEOUV' (LAMBDA ()  
      (PLUSPETIT (PRESV VOLV NV) PRES)))
```

FONCTION CALCULANT LE DEBIT PASSANT PAR LA VALVE MITRALE

```
53. (DEF 'DEBITMIT' (LAMBDA ()  
      (COND ( (MITRALEOUV())  
              (DIV (DIFF PRES (PRESV VOLV NV))  
                    RMITRALEPAS))  
            (T (DIV (DIFF PRES (PRESV VOLV NV))  
                  RMITRALEBLOC)))))
```

FONCTION CALCULANT LE DEBIT PASSANT PAR LA VALVE AORTIQUE

a) dans le contexte de l'aorte

```
54. (DEF 'DEBITAOCONTAO' (LAMBDA ()  
      (COND ( (AORTIQUEOUV())  
              (SETQ DAO (DIV (DIFF (CONVERSIONPV()) PSYS)  
                              (ADD RAORTIQUEPAS RCAR)))  
            (COND ( (PLUSPETIT DAO (VOLV PV NV)) DAO)  
                  (T (VOLV PV NV)) ) )  
            (T (DIV (DIFF (CONVERSIONPV()) (PRESAO VOLAO RAID))  
                  RAORTIQUEBLOC)))))
```

b) dans le contexte du ventricule

```
55. (DEF 'DEBITAOCONTV' (LAMBDA ()  
      (COND ( (AORTIQUEOUV())  
              (SETQ DAO (DIV (DIFF (PRESV VOLV NV) PSYS)  
                              (ADD RAORTIQUEPAS RCAR)))  
            (COND ( (PLUSPETIT DAO (VOLV PV NV)) DAO)  
                  (T (VOLV PV NV)) ) )  
            (T (DIV (DIFF (PRESV VOLV NV) (CONVERSIONPA()))  
                  RAORTIQUEBLOC)))))
```

FONCTION CALCULANT LE VOLUME DANS LE VENTRICULE

```
56. (DEF 'VOLV' (LAMBDA (PV NV)  
      (COND ((NULL PV) VOLV)  
            (T (DIVV PV NV)))))
```



FONCTION CALCULANT LE VOLUME DDANS L'AORTE

```
57. (DEF 'VOLAO '(LAMBDA (PA RAID)
      (COND ((NULL PA) VOLAO)
      (T (DIVAO PA RAID) ))))
```

FONCTION CALCULANT LE DEBIT VERS LE RESEAU PERIPHERIQUE

```
58. (DEF 'DEBITPER '(LAMBDA ()
      (DIV (PRESAO VOLAO RAID) RPER)))
```

FONCTION DETERMINANT LA PHASE DANS LAQUELLE SE TROUVE LE COEUR

```
59. (DEF 'PHASE '(LAMBDA ()
      (COND ((EQUAL AUGMNV '0) (SETQ P 'REPOS))
      ((AND (NOT (AORTIQUEOUV())) (EQUAL AUGMNV '+))
        (SETQ P 'CONTRAC-ISO))
      ((AND (AORTIQUEOUV()) (EQUAL AUGMNV '+))
        (SETQ P 'CONTRAC-EJEC))
      ((EQUAL AUGMNV '-') (SETQ P 'RELAXATION)))))
```

FONCTION CALCULANT LA VARIATION DU VOLUME DE SANG DANS LE VENTRICULE

```
60. (DEF 'VARIATIONVOLV '(LAMBDA ()
      (DIFF (DEBITMIT ()) (DEBITAOCONTV ()) )))
```

FONCTION CALCULANT LA VARIATION DU VOLUME DE SANG DANS L'AORTE

```
61. (DEF 'VARIATIONVOLAO '(LAMBDA ()
      (DIFF (DEBITAOCONTAO ()) (DEBITPER ()) )))
```

FONCTION PROPAGEANT LA VALEUR DU VOLUME DE SANG DANS LE VENTRICULE

```
62. (DEF 'VOLVAPRES '(LAMBDA ()
      (SETQ VOLVT (ADDVAP VOLV (MULTVAP
        (VARIATIONVOLV ()) (DT ()))))))
```

FONCTION PROPAGEANT LA VALEUR DU VOLUME DE SANG DANS L'AORTE

```
63. (DEF 'VOLAOAPRES '(LAMBDA ()
      (SETQ VOLAOT (ADDAOAP VOLAO (MULTAOAP
                                   (VARIATIONVOLAO()) (DT())))))
```

FONCTION EXECUTANT LA PROPAGATION DU TEMPS

```
64. (DEF 'INSTANTAPRES '(LAMBDA ()
      (VOLVAPRES ())
      (VOLAOAPRES ())
      (SETQ VOLV VOLVT)
      (SETQ VOLAO VOLAOT)
      (NVAPRES())))
```

FONCTION DETERMINANT LA LONGUEUR DE L'INTERVALLE A CONSIDERER  
LORS DE LA PROPAGATION DANS LE TEMPS

```
65. (DEF 'DT '(LAMBDA ()
      (COND ((EQUAL RYTHMEC 'NORM)
              (RNORM()))
            ((EQUAL RYTHMEC 'COURT)
              (RCOURT()))
            ((EQUAL RYTHMEC 'LONG)
              (RLONG())))))
```

```
66. (DEF 'RNORM '(LAMBDA ()
      (COND ((EQUAL REPOS 'DEBUT) 'G)
            ((EQUAL REPOS 'FIN) 'TP)
            ((EQUAL AUGMNV '+) 'TP)
            ((EQUAL AUGMNV '-')
              (COND ((EQUAL EMAX 'NORM)
                      (COND ((EQUAL NV 'P) 'TP)
                            ((EQUAL NV 'MG) 'P)))
                    ((EQUAL EMAX 'GRAND)

                      (COND ((EQUAL NV 'MP) 'TP)
                            ((EQUAL NV 'G) 'P)))
                    ((EQUAL EMAX 'PETIT)
                      (COND ((EQUAL NV 'TP) 'TP)
                            ((EQUAL NV 'MP) 'P)))))))
```



```

67. (DEF 'RCOURT ' (LAMBDA ()
      (COND ((NOT (EQUAL REPOS 'DEBUT)) 'TP)
              (T 'MG))))

```

```

68. (DEF 'RLONG ' (LAMBDA ()
      (COND ((EQUAL AUGMNV '+) 'P)
              ((EQUAL REPOS 'FIN) 'MP)
              ((EQUAL REPOS 'DEBUT) 'TG)
              ((EQUAL AUGMNV '-')
                (COND ((EQUAL EMAX 'NORM)
                        (COND ((EQUAL NV 'P) 'P)
                                ((EQUAL NV 'MG) 'MP)))
                        ((EQUAL EMAX 'GRAND)
                        (COND ((EQUAL NV 'MP) 'P)
                                ((EQUAL NV 'G) 'MP)))
                        ((EQUAL EMAX 'PETIT)
                        (COND ((EQUAL NV 'TP) 'P)
                                ((EQUAL NV 'MP) 'MP)))))))

```

FONCTION DETERMINANT LE NOMBRE D'ELASTIQUES AUTOUR DU VENTRICULE  
LORS DE LA PROPAGATION DANS LE TEMPS

```

69. (DEF 'NVAPRES ' (LAMBDA ()
      (COND ((AND (EQUAL REPOS 'DEBUT) (EQUAL NV 'O))
              (SETQ REPOS 'FIN) NV)
              ((EQUAL EMAX 'NORM)
                (COND ((EQUAL AUGMNV '-')
                        (COND ((EQUAL NV 'P) (SETQ AUGMNV 'O)
                                (SETQ REPOS 'DEBUT)
                                NV)
                                ((EQUAL NV 'MG) (SETQ NV 'P))))
                        ((AND (EQUAL REPOS 'FIN) (EQUAL NV 'O))
                        (SETQ REPOS NIL)
                        (SETQ AUGMNV '+)
                        (SETQ NV 'P))
                        ((EQUAL AUGMNV '+)
                        (COND ((EQUAL NV 'P) (SETQ NV 'MP))
                                ((EQUAL NV 'MP) (SETQ NV 'MG))

                                ((EQUAL NV 'MG) (SETQ NV 'G))
                                ((EQUAL NV 'G) (SETQ AUGMNV '-')
                                (SETQ EJEC NIL)
                                (SETQ NV 'MG))))))

```

```

((EQUAL EMAX 'GRAND)
  (COND ((EQUAL AUGMNV '-')
    (COND ((EQUAL NV 'MP) (SETQ AUGMNV '0)
      (SETQ REPOS 'DEBUT)
      (SETQ NV '0))
      ((EQUAL NV 'G) (SETQ NV 'MP))))
    ((AND (EQUAL REPOS 'FIN) (EQUAL NV '0))
      (SETQ REPOS NIL)
      (SETQ AUGMNV '+)
      (SETQ NV 'MP))
    ((EQUAL AUGMNV '+)
      (COND ((EQUAL NV 'MP) (SETQ NV 'MG))
        ((EQUAL NV 'MG) (SETQ NV 'G))
        ((EQUAL NV 'G) (SETQ NV 'TG))
        ((EQUAL NV 'TG) (SETQ AUGMNV '-')
          (SETQ EJEC NIL)
          (SETQ NV 'G))))))
  ((EQUAL EMAX 'PETIT)
    (COND ((AND (EQUAL REPOS 'FIN) (EQUAL NV '0))
      (SETQ REPOS NIL)
      (SETQ AUGMNV '+)
      (SETQ NV 'TP))
    ((EQUAL AUGMNV '-')
      (COND ((EQUAL NV 'TP) (SETQ AUGMNV '0)
        (SETQ REPOS 'DEBUT)
        (SETQ NV '0))
        ((EQUAL NV 'MP) (SETQ NV 'TP))))
    ((EQUAL AUGMNV '+)
      (COND ((EQUAL NV 'TP) (SETQ NV 'P))
        ((EQUAL NV 'MP) (SETQ NV 'MG))
        ((EQUAL NV 'MG) (SETQ AUGMNV '-')
          (SETQ EJEC NIL)
          (SETQ NV 'MP))))))

```